

UNIVERZITA PARDUBICE
DOPRAVNÍ FAKULTA JANA PERNERA

DIPLOMOVÁ PRÁCE

2015

Bc. Tomáš Nadrchal

Univerzita Pardubice

Dopravní Fakulta Jana Pernera

Pravděpodobnostní kalkulátor

Tomáš Nadrchal

Diplomová práce

2015

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Tomáš Nadrchal**
Osobní číslo: **D13660**
Studijní program: **N3708 Dopravní inženýrství a spoje**
Studijní obor: **Aplikovaná informatika v dopravě**
Název tématu: **Pravděpodobnostní kalkulátor**
Zadávající katedra: **Katedra informatiky v dopravě**

Z á s a d y p r o v y p r a c o v á n í :

Cílem diplomové práce je zdokonalení a rozšíření programového nástroje pro generování hodnot náhodné proměnné, který student vytvořil v rámci bakalářské práce. Budou doplněny generátory náhodných čísel pro další typy náhodných proměnných, nově budou vytvořeny aplikace umožňující prokládání naměřených dat vybranými typy teoretických rozdělení pravděpodobnosti a testování náhodnosti v datových souborech. Výstupem práce bude samostatně běžící aplikace napsaná v programovém prostředí C#.

Rozsah grafických prací:

Rozsah pracovní zprávy: **40 normostran**

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

BANKS Jerry. Discrete-event system simulation. 3rd ed. Upper Saddle River Prentice Hall, 2001

FORBES, Catherine, Merran EVANS, Nicholas HASTINGS a Brian PEACOCK. Statistical Distributions. 4th ed. Hoboken, New Jersey: Wiley, 2011, xviii, 212 p. ISBN 978-047-0390-634.

HUŠEK, Roman a Josef LAUBER. Simulační modely. Praha: SNTL - Nakladatelství technické literatury, 1987. DT 330.116.1(075.8).

Vedoucí diplomové práce:

Mgr. Věra Záhorová, Ph.D.

Katedra informatiky v dopravě

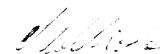
Datum zadání diplomové práce: **5. prosince 2014**

Termín odevzdání diplomové práce: **22. května 2015**



doc. Ing. Ivo Drahotský, Ph.D.
děkan

L.S.



doc. Ing. Vladimír Jehlička, CSc.
vedoucí katedry

V Pardubicích dne 5. prosince 2014

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 1. 5. 2015

Bc. Tomáš Nadrchal

PODĚKOVÁNÍ

Tímto bych chtěl poděkovat paní Věře Záhorové, Ph.D. za vedení práce a panu doc. Ing. Karlu Greinerovi, Ph.D. za přínosné konzultace.

ANOTACE

Výsledná aplikace se zabývá generováním hodnot náhodné proměnné, výpočty a porovnáváním charakteristik implementovaných rozdělení pravděpodobnosti, testováním náhodných čísel, analýzou naměřených dat a umožňuje výpočet funkčních hodnot některých relativně složitých funkcí. Navíc je doplněna o publikaci, která obsahuje potřebný teoretický aparát, podrobný návod k ovládání aplikace a popis její struktury. Díky velkému množství rozdělení pravděpodobnosti lze generátory čísel a možnost analýzy dat využít nejen v dopravních simulačních modelech, ale napříč všemi obory. Své místo si práce najde i na akademické půdě, kde může pomoci nejen studentům teorie pravděpodobnosti a matematické statistiky.

KLÍČOVÁ SLOVA

Generátory náhodných čísel, rozdělení pravděpodobnosti, náhodná veličina, náhodná čísla, pseudonáhodná čísla, testy náhodných čísel, analýza dat

TITLE

Probability calculator

ANNOTATION

Final application deals with a generation of values of random variable, calculations and comparisons characteristics of implemented probability distributions, random number tests, analysis of measured data and allows a calculation functional values of relatively complex functions. It is also supplemented by a publication containing necessary theoretical basis, detailed instructions to operate the application and a description of its structure. Due to the large amount of probability distribution, number generators and a possibility of data analysis can be used not only in traffic simulation models but perhaps across all branches. This application finds its place also in academic campus where can be useful not only for students of probability theory and mathematical statistics.

KEYWORDS

Random number generators, probability distributions, random variable, random number, pseudorandom number, random number tests, data analysis

OBSAH

0	ÚVOD	13
1	TEORETICKÝ APARÁT	14
1.1	Generování náhodných čísel.....	14
1.1.1	Náhodná čísla.....	14
1.1.2	Získávání náhodných čísel.....	14
1.1.3	Lineární kongruenční generátor.....	15
1.1.4	Kvadratický kongruenční generátor.....	17
1.1.5	Další metody	17
1.2	Testování náhodných čísel.....	17
1.2.1	Testy shody	18
1.2.2	Mezerový test.....	19
1.2.3	Test výskytu úplných sad číslic	20
1.2.4	Test autokorelace	21
1.2.5	Úsekový test.....	21
1.2.6	Poker test.....	22
1.2.7	Frekvenční test.....	23
1.2.8	Vizuální analýza	24
1.3	Vybraná spojitá rozdělení pravděpodobnosti	24
1.3.1	Arcsinové rozdělení	25
1.3.2	Rozdělení extrémních hodnot	26
1.3.3	Gumbelovo rozdělení pravděpodobnosti	26
1.3.4	Chí-kvadrát rozdělení	27
1.3.5	Kosinové rozdělení	28
1.3.6	Logaritmické rozdělení	29
1.3.7	Maxwellovo rozdělení	30
1.3.8	Paretovo rozdělení	31

1.4	Vybraná diskrétní rozdělení pravděpodobnosti	32
1.4.1	Bernoulliho rozdělení	32
1.4.2	Binomické rozdělení	33
1.4.3	Geometrické rozdělení	33
1.4.4	Hypergeometrické rozdělení	34
1.4.5	Negativně binomické rozdělení	35
1.4.6	Pascalovo rozdělení	36
1.4.7	Poissonovo rozdělení	37
1.4.8	Rovnoměrné rozdělení	38
1.5	Transformace náhodných čísel	39
1.5.1	Metoda inverzní transformace	39
1.5.2	Zamítací metoda	40
1.5.3	Kompoziční metoda	41
1.5.4	Další možnosti transformace	41
1.6	Speciální funkce	42
1.6.1	Faktoriál	42
1.6.2	Kombinační číslo	43
1.6.3	Stirlingovo číslo druhého druhu	43
1.6.4	Chybová funkce	43
1.6.5	Gama funkce	44
1.6.6	Beta funkce	44
1.6.7	Neúplná gama a beta funkce	44
2	OVLÁDÁNÍ APLIKACE	45
2.1	Generátor	46
2.2	Charakteristiky	48
2.3	Porovnávání	50
2.4	Testování náhodných čísel	51

2.5	Analýza dat	53
2.6	Další funkce	54
2.7	Kritické hodnoty	56
2.8	Nastavení	57
2.8.1	Generátor	57
2.8.2	Graf	58
2.8.3	Histogram.....	59
2.8.4	Testy náhodnosti	59
2.8.5	Zaokrouhlování	60
2.8.6	Barvy.....	61
3	STRUKTURA APLIKACE.....	62
3.1	Control	62
3.1.1	Generátory pseudonáhodných čísel	62
3.1.2	Testy náhodných čísel.....	63
3.1.3	Transformace náhodných čísel	64
3.1.4	Charakteristiky.....	65
3.1.5	Matematika	65
3.2	Entity	66
3.2.1	Histogram.....	66
3.2.2	Výčtové typy.....	67
3.2.3	Nastavení	68
3.3	Presentation	68
3.3.1	Formuláře.....	68
3.3.2	Textové soubory	68
4	ZÁVĚR	69
5	POUŽITÁ LITERATURA	70
6	PŘÍLOHY	72

SEZNAM ILUSTRACÍ

Obr. 1: Algoritmus získání dat pro mezerový test [3]	19
Obr. 2: Princip úsekového testu	21
Obr. 3: Vizuální analýza	24
Obr. 4: Hustota pravděpodobnosti a distribuční funkce arcsinova rozdělení	25
Obr. 5: Hustota pravděpodobnosti a distribuční funkce rozdělení extrémních hodnot	26
Obr. 6: Hustota pravděpodobnosti a distribuční funkce Gumbelova rozdělení	27
Obr. 7: Hustota pravděpodobnosti a distribuční funkce chí-kvadrát rozdělení	28
Obr. 8: Hustota pravděpodobnosti a distribuční funkce kosinova rozdělení	29
Obr. 9: Hustota pravděpodobnosti a distribuční funkce logaritmického rozdělení	30
Obr. 10: Hustota pravděpodobnosti a distribuční funkce Maxwellova rozdělení	31
Obr. 11: Hustota pravděpodobnosti a distribuční funkce Paretova rozdělení	31
Obr. 12: Pravděpodobnostní funkce a distribuční funkce Bernoulliho rozdělení	32
Obr. 13: Pravděpodobnostní funkce a distribuční funkce binomického rozdělení	33
Obr. 14: Pravděpodobnostní funkce a distribuční funkce geometrického rozdělení	34
Obr. 15: Pravděpodobnostní funkce a distribuční funkce hypergeometrického rozdělení	35
Obr. 16: Pravděpodobnostní funkce a distribuční funkce negativně binomického rozdělení	36
Obr. 17: Rozdíl mezi negativně binomickým a Pascalovým rozdělením	36
Obr. 18: Pravděpodobnostní funkce a distribuční funkce Pascalova rozdělení	37
Obr. 19: Pravděpodobnostní funkce a distribuční funkce Poissonova rozdělení	38
Obr. 20: Pravděpodobnostní funkce a distribuční funkce rovnoměrného rozdělení	39
Obr. 21: Princip kompoziční metody [3]	41
Obr. 22: Hlavní okno aplikace	45
Obr. 23: Karta generování hodnot náhodné proměnné	46
Obr. 24: Uložení grafického výstupu	47
Obr. 25: Ukládání vygenerovaných hodnot	48
Obr. 26: Karta charakteristik náhodné proměnné	49
Obr. 27: Karta porovnávání náhodné proměnné	50
Obr. 28: Karta testování náhodné proměnné	52
Obr. 29: Karta analýzy dat	53
Obr. 30: Karta s dalšími funkcemi	55
Obr. 31: Karta s kritickými hodnotami	56
Obr. 32: Kontextová nabídka menu pro nastavení	57

Obr. 33: Dialogové okno pro nastavení generátoru	58
Obr. 34: Formát souboru náhodných čísel	58
Obr. 35: Dialogové okno pro nastavení grafu.....	59
Obr. 36: Dialogové okno pro nastavení tříd histogramu.....	59
Obr. 37: Dialogové okno pro nastavení testů náhodnosti	60
Obr. 38: Dialogové okno pro nastavení zaokrouhlování	60
Obr. 39: Dialogové okno pro nastavení základních barev	61
Obr. 40: Zjednodušený diagram tříd – generátory pseudonáhodných čísel.....	63
Obr. 41: Zjednodušený diagram tříd – testy náhodných čísel	63
Obr. 42: Zjednodušený diagram tříd – transformace náhodných čísel	64
Obr. 43: Zjednodušený diagram tříd – charakteristiky náhodné proměnné.....	65
Obr. 44: Zjednodušený diagram třídy Matematika.....	66
Obr. 45: Zjednodušený diagram tříd – histogram	67
Obr. 46: Zjednodušený diagram tříd – výčtové typy	68
Obr. 47: Diagram třídy pro práci s textovými soubory.....	68

SEZNAM TABULEK

Tab. 1: Poker test – varianty pětic [2]	22
Tab. 2: Modifikovaný poker test – varianty pětic [2]	23
Tab. 3: Modifikovaný poker test – varianty pětic s pravděpodobnostmi.....	23

0 ÚVOD

Tématem diplomové práce je rozšíření programového nástroje vytvořeného v rámci bakalářského studia o nová rozdělení pravděpodobnosti a další funkcionality. Výstupem je pak samostatně běžící aplikace, která disponuje celkem 31 různými spojitými i diskrétními rozděleními pravděpodobnosti. Můžeme tak nejen generovat hodnoty náhodné proměnné těchto rozdělení, ale také zkoumat jejich vlastnosti a charakteristiky. Zachována je samozřejmě i možnost jednotlivá rozdělení porovnávat mezi sebou.

Aplikace byla navíc doplněna o nové funkce. Patří mezi ně například možnost testování náhodnosti. Zde můžeme prověřit buď vlastní hodnoty, nebo jeden ze tří implementovaných generátorů, u kterých máme možnost nastavení vlastních parametrů. K tomuto účelu jsou implementovány 2 vizuální a 6 empirických testů.

Další novinkou je analýza naměřených údajů. Po vložení souboru hodnot jsou vypočítány odhady jeho základních charakteristik. Poté je může uživatel proložit libovolným teoretickým rozdělením pravděpodobnosti. Nejen že je proložení vykresleno do histogramu, ale jsou provedeny i základní testy shody, konkrétně χ^2 a Kolmogorovův-Smirnovův test. Aplikace nově nabízí také tabulky kritických hodnot nutných pro statistické testy a možnost výpočtu některých zajímavých a celkem složitých funkcí. Za zmínění stojí například gama a beta funkce, chybová funkce i výpočet hodnoty Stirlingova čísla druhého druhu.

I přes intuitivní a uživatelsky přívětivé rozhraní je nutné pro práci s aplikací disponovat alespoň základními znalostmi dané problematiky. Aby uživatel nemusel prostudovat desítky knih, je práce doplněna o dokumentaci obsahující i kapitolu s potřebným teoretickým aparátem. Další části jsou pak věnovány návodu k ovládání nástroje a jeho vnitřní struktuře. Mou snahou bylo sepsání relativně složitých témat čtivou a především srozumitelnou formou.

Mým hlavním cílem bylo vytvoření užitečného nástroje, který bude možné použít v dlouhodobém časovém horizontu a ne pouze jako prostředek k získání vysokoškolského diplomu. Široké uplatnění tak aplikace nachází například v simulačních modelech, a to hned v několika fázích jejich životního cyklu. Na začátku můžeme analyzovat naměřená data a určit jejich teoretické rozdělení pravděpodobnosti. Na základě této informace jsme schopni vygenerovat hodnoty vstupních proudů simulačního modelu, a tak zvýšit jeho kredibilitu. A nakonec máme možnost vyhodnotit i výstupní údaje modelu.

Krom výše zmíněného najde tento nástroj své uplatnění i na akademické půdě, kde může být využit jako studijní opora nejen při studiu teorie pravděpodobnosti či matematické statistiky.

1 TEORETICKÝ APARÁT

Aby mohl uživatel naplno využívat přiloženou aplikaci, musí se alespoň na základní úrovni orientovat v dané problematice. A právě k tomu je určena tato kapitola.

Vzhledem k náročnosti některých témat je předpokládána základní znalost teorie pravděpodobnosti a matematické statistiky. Všechny potřebné informace však čtenář nalezne v mé bakalářské práci [1], na kterou tato publikace navazuje.

1.1 Generování náhodných čísel

První stěžejní částí je generování náhodných čísel. Nejdříve si připomeneme, co to náhodná čísla vůbec jsou, k čemu slouží a jak je možné je získat. Na konci kapitoly si ukážeme dva způsoby samotného generování.

1.1.1 Náhodná čísla

Pod pojmem náhodná čísla budeme rozumět hodnoty náhodné veličiny s rovnoměrným rozdělením na intervalu $(0; 1)$. Aby mohla být posloupnost čísel označena jako náhodná, musí splňovat určitá kritéria.

Prvním základním požadavkem je nezávislost jednotlivých prvků. To ve zkratce znamená, že pokud z posloupnosti vybereme jedno číslo, nedokážeme odhadnout, jaké číslo bude následovat. Dalším důležitým kritériem je rovnoměrné rozložení hodnot po celém intervalu. Obdobných pravidel je však celá řada a podrobněji se jimi budeme zabývat v kapitole o testování náhodných čísel.

K čemu jsou taková náhodná čísla potřeba? Využit se dají například v simulacích, kdy je pak možné napodobit chování reálných systémů, například příchod cestujících na zastávku či dobu zpoždění vlaku. Další uplatnění nacházejí v kryptografii, kde můžeme za pomoci náhodnosti ochránit citlivá data nebo šifrovat komunikaci. Pomocí sekvence náhodných údajů jsme schopni otestovat chování aplikací ještě před uvedením do provozu a zajistit tak stabilní chod i v extrémních podmínkách. To je ovšem jen zlomek ze všech možností jejich nasazení. Další příklady čtenář nalezne ve zdroji [2] a [3].

V mnoha aplikacích je nutné náhodné číslo s rovnoměrným rozdělením transformovat na číslo s jiným rozdělením pravděpodobnosti. Tím se zabývá samostatná kapitola této práce.

1.1.2 Získávání náhodných čísel

Nyní se podíváme, jak je možné náhodná čísla získat. Ve své bakalářské práci [1] jsem již zmínil, že náhodné hodnoty můžeme dostat hned několika způsoby. Ty si zde však pouze připomeneme a zaměříme se jen na stěžejní metody.

Jako první uvedeme **tabulky náhodných čísel**. Jejich předností je zaručená náhodnost. Nevýhodou pak počet těchto čísel, který je například pro simulační modely nedostatečný. Zmínit můžeme Tippetovy tabulky z roku 1927 s 40 000 čísly nebo tabulky RAND Corp. z roku 1955 s již 1 milionem číslic.[1]

Další možností jsou **fyzikální generátory** náhodných čísel. Ty jsou založeny na registraci určitých fyzikálních pochodů, například měření délky intervalů mezi dopady částic na registrační plochu. Jako nevýhodu je třeba zmínit nemožnost identického opakování fyzikálního procesu.[1]

A konečně se dostáváme k **aritmetickým algoritmům**, na které se podíváme podrobněji. Ty vytvářejí náhodná čísla na základě jednoduchých rekurentních výpočtů, v nichž následující číslo deterministicky závisí na jednom či více předchozích.

Pozorný čtenář by nyní namítl, že tato čísla pak nemohou být označena jako náhodná, protože nesplňují ani základní pravidla náhodnosti. Tím se dostáváme k dalšímu důležitému pojmu, **pseudonáhodná čísla**.

Pseudonáhodná čísla mohou být produktem právě aritmetických algoritmů. Nesplňují sice všechna pravidla náhodnosti, ale jedná-li se o kvalitní algoritmus, vygenerovaná čísla se k těm náhodným velice přibližují. K této problematice se dostaneme v již zmíněné kapitole o testování náhodných čísel.

Historicky prvním aritmetickým algoritmem byla von Neumannova metoda „prostřednictvím řádů druhé mocniny“. Byla navržena již v roce 1946 a měla celou řadu nedostatků. Tím hlavním byla velice krátká perioda, tedy malý počet různých hodnot předtím, než došlo k jejich opakování. K tomuto nepříjemnému jevu docházelo při zvolení špatného počátečního čísla.[1]

Dnes nejpoužívanější generátory fungují na principu založeném Lehmerem již v roce 1949 a jsou označovány jako **lineárně kongruenční**. [1]

1.1.3 Lineární kongruenční generátor

Požadovanou posloupnost pseudonáhodných čísel získáme pomocí aritmetického rekurentního předpisu.

$$x_{n+1} = (ax_n + c) \bmod m, \quad n \geq 0$$

Aby se získané hodnoty co nejvíce přibližovaly pravidlům náhodnosti, je potřeba vhodně zvolit konstanty m , a , c a násadu x_0 . Právě zvolením těchto parametrů můžeme podstatně ovlivnit kvalitu výsledného generátoru. Operátor mod představuje tzv. modulo,

zbytek po celočíselném dělení. Proměnná x_n vyjadřuje n -tý člen posloupnosti vygenerovaných čísel.

Prvním zmíněným parametrem je **modul**, či modulus m a platí pro něj, že $m > 0$. Jako jeho hodnotu je vhodné zvolit poměrně velké číslo, závisí na ní totiž délka periody, ta nemůže mít větší délku než je m .

Dále je potřeba volit modul tak, aby samotný výpočet probíhal co nejrychleji. První možností je zvolit m jako prvočíslo. To ale vyžaduje určení množiny přípustných hodnot parametru a . Druhou alternativou pak může být určení modulu jako mocniny dvou.[4]

Jako další parametr zmíníme **multiplikativní konstantu**, neboli násobitel a , pro který platí $0 \leq a < m$.

Následuje parametr c , kde $0 \leq c < m$, jenž je označován jako **aditivní konstanta** nebo inkrement. Při zvolení $c = 0$ dochází k o něco rychlejšímu generování čísel, ale nikdy tak nedosáhneme maximální periody.[4]

Poslední hodnotou je tzv. **násada**, tedy počáteční hodnota x_0 . Platí pro ni $0 \leq x_0 < m$ a umožňuje nám znovu reprodukovat vygenerovaný proud čísel. Někdy bývá označována také jako semínko. I zde ji můžeme volit několika způsoby. Pro zajištění již zmíněné reprodukovatelnosti volíme konkrétní hodnotu. Jinak můžeme použít například aktuální systémový čas či hodnotu z libovolného generátoru čísel.

Nejlepších výsledků dosáhneme použitím vhodné kombinace všech parametrů. Podrobný popis, jak při výběru postupovat, čtenář nalezne ve zdroji [3]. Nyní si ukážeme příklad generování čísel uvedený ve zdroji [5]. Předpis pro výpočet n -tého členu je oproti výše uvedenému modifikován, význam symbolů je však stejný. Tento předpis je použit pro generování i v přiložené aplikaci.

$$x_{n+1} = (ax_n - 1 + c) \bmod m$$

Protože x_i jsou celá čísla, takže nenáleží do intervalu $(0; 1)$, musí být následně ještě upravena. Výsledné prvky posloupnosti vygenerovaných čísel tedy označíme u_i a získáme je pomocí následujícího vzorce.

$$u_i = \frac{x_i}{m}$$

Nyní zvolíme vstupní parametry, a to tak, že $m = 2^{31} - 1 = 2\,147\,483\,647$, $c = 0$, $a = 16\,807$ a $x_0 = 12\,345$.

$$x_1 = (16\,807 \cdot 12\,345) \bmod m = 207\,482\,115$$

$$u_1 = \frac{207\,482\,115}{m} = 0,096\,616\,528\,5$$

1.1.4 Kvadratický kongruenční generátor

Předpis kvadratického kongruenčního generátoru patří mezi nelineární způsoby získávání pseudonáhodných čísel. V tomto případě se vlastně jedná pouze o zobecnění výše zmíněné lineární kongruenční metody.[3]

$$x_{n+1} = (ax_n^2 + bx_n + c) \bmod m$$

Oproti lineárnímu kongruenčnímu předpisu nám přibyl parametr b , který představuje **multiplikativní konstantu lineárního členu**. Konstantě a zůstává funkce násobitele, konkrétně se pak jedná o **multiplikativní konstantu kvadratického členu**. Zbývající parametry mají obdobný význam jako u předchozího generátoru.

1.1.5 Další metody

Metod získávání pseudonáhodných čísel je nepřehledné množství. Každá má své přednosti, ale i nedostatky. Při výběru generátoru je vhodné vzít v úvahu i účel jeho nasazení. Některé jsou vhodné pro simulační modely, jiné zase pro použití v kryptografii. Při hlubším zájmu může čtenář využít publikace [3], [4] a [6].

1.2 Testování náhodných čísel

K objektivnímu zhodnocení kvality generátoru můžeme využít celou řadu metod testování. Základní dělení a letný popis některých způsobů je uveden již v mé bakalářské práci [1]. My se zde na toto téma však podíváme podrobněji, ukážeme si různé metody ověřování náhodnosti a vysvětlíme si princip některých testů.

Testy se dělí na teoretické a empirické. **Teoretické** k ověřování náhodnosti využívají odvětví matematiky zvané teorie čísel. Jejich závěry jsou platné pro celou periodu. **Empirické** testy pak slouží k ověření kritérií náhodnosti vygenerované posloupnosti čísel. V tomto případě dochází k vyhodnocení předpokládaných a naměřených statistik.[1, 2]

Těchto kritérií lze však formulovat velmi mnoho, a tak je potřeba hned na začátku upozornit na fakt, že i když náš generátor projde v n testech, nemůžeme s jistotou říci, že uspěje i v testu $n + 1$. V praxi se tak doporučuje použít 5-6 testů. Pokud jimi čísla projdou, můžeme generátor považovat za vyhovující.[1, 2]

Nejdříve si tedy osvěžíme znalosti a zmíníme testy dobré shody, konkrétně χ^2 a Kolmogorovův-Smirnovův test. Dále se budeme věnovat některým empirickým testům. Ukážeme si, jak fungují a jak je vyhodnotit. Nakonec si ukážeme další zajímavou možnost testování, jež sice není úplně objektivní, ale i tak nám může při posuzování kvality generátoru dobře posloužit.

1.2.1 Testy shody

Název testy shody se používá pro označení skupiny testů, jež slouží k testování hypotéz o tvaru rozdělení pravděpodobnosti. Obvykle se testuje nulová hypotéza H_0 , která říká: „Výběr pochází z daného rozdělení pravděpodobnosti s danými parametry.“, oproti alternativní hypotéze H_1 : „Nulová hypotéza neplatí.“. Některé testy slouží přímo pro konkrétní typy rozdělení. Existuje například celá řada testů normálního rozdělení. My se ovšem budeme zabývat těmi univerzálními.[7]

Testy shody tedy nejsou přímo testy náhodnosti, ale často se používají pro jejich vyhodnocení. Použít je lze i při kontrole hodnot náhodné proměnné, kdy je možné vygenerované hodnoty porovnat s požadovaným teoretickým rozdělením pravděpodobnosti.

Nejdříve se zaměříme na **chí-kvadrát test**, který je snad nejznámějším ze všech statistických testů shody a často se používá právě k vyhodnocení mnoha jiných testů. Je vhodný zejména v případech, kdy máme dostatečný počet dat. Aby měl test dobrou vypovídající hodnotu, mělo by být ve většině tříd alespoň 5 prvků. Ve všech třídách pak musí být minimálně jeden prvek. V opačném případě může dojít ke zbytečnému zamítnutí platné nulové hypotézy. Použít jej dále nemůžeme v případech, kdy je mezi naměřenými prvky nějaká závislost.[3, 7]

Nejdříve rozdělíme data do k vhodných tříd a poté pomocí χ^2 testu sledujeme rozdíl mezi zjištěnými četnostmi n_i^e v jednotlivých třídách a četnostmi n_i^t , které bychom v daných třídách očekávali v případě platnosti nulové hypotézy.[7]

Výsledkem pak je testovací kritérium, které se řídí χ^2 rozdělením pravděpodobnosti s $k - r - 1$ stupni volnosti, kde k je počet tříd a r počet odhadovaných parametrů. Hodnotu testovacího kritéria získáme z níže uvedeného vztahu.[7]

$$R = \sum_{i=1}^k \frac{(n_i^e - n_i^t)^2}{n_i^t}$$

V případě platnosti nulové hypotézy by se četnosti n_i^e a n_i^t v jednotlivých třídách musely rovnat a celková hodnota testovacího kritéria by musela být rovna nule. Pokud přijímáme rozhodnutí na hladině významnosti α , nulovou hypotézu zamítáme v případě, kdy je hodnota testovacího kritéria větší než kvantil $\chi_{1-\alpha; k-r-1}^2$. [7]

Pokud nastane případ, že nemáme dostatečné množství dat, můžeme použít **Kolmogorovův-Smirnovův test**. Ten funguje na principu porovnávání hodnot empirické distribuční funkce, kterou odhadujeme pomocí relativní kumulativní četnosti, a distribuční funkce testovaného rozdělení pravděpodobnosti v bodech, jež odpovídají hodnotám výběrového souboru. Testovacím kritériem je tzv. supremum, tedy největší ze zjištěných rozdílů. K vyhodnocení testů slouží speciální tabulka s kritickými hodnotami. Kritický obor pak tvoří hodnoty větší než příslušná kritická hodnota.[7] Na konečném souboru hodnot můžeme pro nalezení testovacího kritéria použít následující předpis.

$$R = \max_{1 \leq i \leq n} \left(\left| \frac{i}{n} - F(x_i) \right|; \left| F(x_i) - \frac{i-1}{n} \right| \right)$$

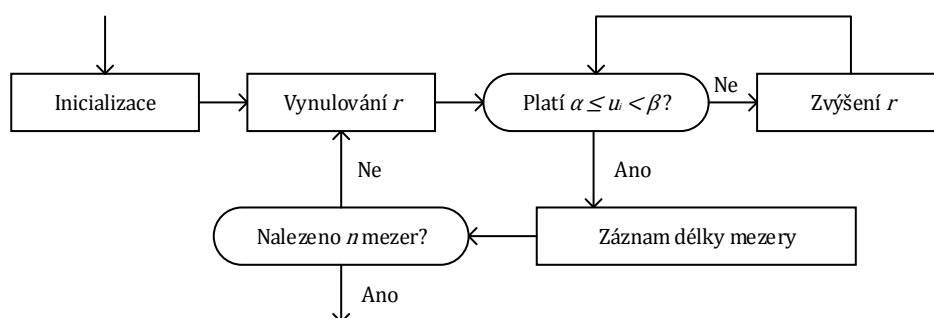
Nevýhodou testu je menší schopnost zamítnout neplatnou nulovou hypotézu, a to zejména v případech, kdy máme k dispozici opravdu malý počet hodnot.[7]

Nyní se blíže podíváme na empirické testy náhodnosti. Ty zkoumají, zda vygenerované či jinak získané posloupnosti náhodných čísel splňují různá kritéria náhodnosti. Hodnoty této posloupnosti označíme jako u_i .

1.2.2 Mezerový test

Tento test, v některých publikacích označovaný jako gap test, zkoumá délku „mezer“ mezi výskyty u_i v jistém intervalu. V našem případě je interval dán reálnými čísly α a β , pro která platí $0 \leq \alpha < \beta \leq 1$. Hledáme tedy délky posloupností $u_i, u_{i+1}, \dots, u_{i+r}$, v nichž u_{i+r} leží mezi α a β , ale ostatní čísla u ne. Tato vybraná posloupnost $r + 1$ čísel pak vyjadřuje délku mezery r . [3]

Nyní si ukážeme, jak získat data pro tento test. Následující algoritmus lze aplikovat na posloupnost hodnot náhodné proměnné, a to pro libovolné parametry α a β . Výstupem jsou počty mezer délky $0, 1, \dots, t - 1$ a délek $\geq t$. Algoritmus je ukončen při nalezení n mezer. [3]



Obr. 1: Algoritmus získání dat pro mezerový test [3]

Jakmile vyhodnotíme vstupní data, můžeme na ně aplikovat χ^2 test s t stupni volnosti, a to s níže uvedenými pravděpodobnostmi jednotlivých tříd.[3]

$$\begin{aligned} p_r &= p(1-p)^r, & 0 \leq r < t \\ p_t &= (1-p)^t, & \text{jinak} \\ p &= \beta - \alpha \end{aligned}$$

Proměnná p vyjadřuje pravděpodobnosti, že $\alpha \leq u_i < \beta$. Hodnoty n a t volíme tak, aby v každé z jednotlivých tříd bylo alespoň 5 mezer, a byly tak splněny požadavky χ^2 testu.[3]

1.2.3 Test výskytu úplných sad čísel

Následující test můžeme najít i pod označením test sběratele kuponů. Jeho úkolem je najít délky takových posloupností U_0, U_1, \dots , které obsahují všechny číslíčky zvolené číselné soustavy o základu d . Posloupnosti dostaneme z hodnot u_i následujícím způsobem.[2]

$$U_i = [u_i d], \quad \text{kde } [] \text{ značí celou část čísla}$$

Poté, obdobně jako u předchozího testu, vytvoříme třídy dle jednotlivých délek posloupností $d, d+1, \dots, t-1$ a $\geq t$. Tady je potřeba upozornit na fakt, že i ta nejkratší posloupnost bude obsahovat minimálně d prvků.

Jakmile vyhodnotíme data a získáme požadovaný počet posloupností n , můžeme využít χ^2 test s $t-d$ stupni volnosti. Pravděpodobnosti jednotlivých kategorií jsou uvedeny níže.[2]

$$\begin{aligned} p_r &= \frac{d!}{d^r} \left\{ \begin{matrix} r-1 \\ d-1 \end{matrix} \right\}, & d \leq r < t \\ p_t &= 1 - \frac{d!}{d^{t-1}} \left\{ \begin{matrix} t-1 \\ d \end{matrix} \right\}, & \text{jinak} \end{aligned}$$

Čísla ve složených závorkách představují Stirlingovo číslo 2. druhu. I zde volíme parametry n a t tak, aby výsledné četnosti splnily požadavky χ^2 testu.

Princip tohoto testu si ukážeme na jednoduchém příkladu. Představme si chlapce, který sbírá kupony d různých typů. Ty mohou být náhodně rozmístěny například v krabicích od oblíbených cereálií. Pokud by chlapec chtěl získat všechny kupony, musel by spořádat alespoň d krabic výrobku.[3] Každý však z vlastní zkušenosti ví, že k zisku všech kuponů je potřeba zakoupit krabic mnohem více.

1.2.4 Test autokorelace

Test autokorelace, případně sériový korelační test, slouží k odhalení případných vazeb mezi prvky. K výpočtu tzv. sériového korelačního koeficientu můžeme použít následující statistiku ze zdroje [3].

$$C = \frac{n(u_0u_1 + u_1u_2 + \dots + u_{n-2}u_{n-1} + u_{n-1}u_0) - (u_0 + u_1 + \dots + u_{n-1})^2}{n(u_0^2 + u_1^2 + \dots + u_{n-1}^2) - (u_0 + u_1 + \dots + u_{n-1})^2}$$

Tento koeficient vyjadřuje, do jaké míry závisí u_{i+1} na u_i . [3] Parametr n zde představuje počet zkoumaných prvků. Jako druhý parametr můžeme zvolit rozestup r mezi dvěma zkoumanými prvky, pro který platí $1 \leq r \leq n - 1$. To v důsledku znamená, že jsou porovnávány prvky u_{i+r} na u_i , což je umožněno i v přiložené aplikaci.

Nyní se ale vrátíme k vyhodnocení testu. Sériový korelační koeficient nabývá hodnot v intervalu $\langle -1; 1 \rangle$. V případě, že je roven nule, nebo se k ní alespoň přibližuje, to znamená, že jsou na sobě prvky posloupnosti relativně nezávislé. Hodnoty ± 1 pak znamenají úplnou lineární závislost. [3]

Žádoucí tedy je, aby se C blížilo k nule. Ve skutečnosti však nelze očekávat, že by mezi prvky žádná korelace nebyla. Musíme tedy určit, kdy je možné nalezenou závislost tolerovat a kdy už ne. Lze předpokládat, že výsledky se řídí normálním rozdělením pravděpodobnosti s parametry μ_n a σ_n .

$$\mu_n = \frac{-1}{n-1}, \quad \sigma_n = \sqrt{\frac{n^2}{(n-1)^2(n-2)}}, \quad n > 2$$

Výsledný koeficient by pak měl v 95% ležet v intervalu $\langle \mu_n - 2\sigma_n; \mu_n + 2\sigma_n \rangle$. Leží-li koeficient C v uvedeném rozmezí, můžeme říct, že posloupnost úspěšně prošla testem autokorelace.

1.2.5 Úsekový test

Úsekových testů je celá řada, v literatuře se vyskytují i pod názvem run testy. My jsme si vybrali úsekový test nad a pod střední hodnotou. Budeme tedy zkoumat délky souvislých úseků pohybujících se pod nebo nad střední hodnotou. Ta je v našem případě 0,5. Pro lepší představu si princip ukážeme na následujícím obrázku.

0,609	0,701	0,359	0,014	0,053	0,844	0,883	0,317	0,691	0,626
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

Obr. 2: Princip úsekového testu

Soubor hodnot má celkem 10 čísel, tedy $n = 10$. Z toho 4 jsou pod a 6 je nad střední hodnotou, takže $n_{pod} = 4$ a $n_{nad} = 6$. Celkový počet úseků je 5, pak $p = 5$. Když už víme, jaké se v testu vyskytují proměnné a známe jejich hodnoty, můžeme je vyhodnotit.

V případě, že $n_{pod} > 20$, můžeme rozdělení testovacího kritéria aproximovat normálním rozdělením pravděpodobnosti.[8] Potřebné parametry μ_n a σ_n pak získáme podle vztahů ze zdroje [8].

$$\mu_n = \frac{1}{2} + \frac{2n_{nad}n_{pod}}{n}, \quad \sigma_n = \sqrt{\frac{2n_{nad}n_{pod}(2n_{nad}n_{pod} - n)}{n^2(n-1)}}$$

Nyní můžeme vypočítat testovací kritérium Z_0 dle zdroje [8].

$$Z_0 = \frac{p - \mu_n}{\sigma_n}$$

Kritický obor pak tvoří hodnoty větší než kvantil normálního normovaného rozdělení na požadované hladině významnosti.

1.2.6 Poker test

Poker test, který můžeme pojmenovat i jako rozkladový, zkoumá frekvence výskytu různých kombinací číslic. My pak můžeme testovat, zda se empiricky zjištěné četnosti těchto kombinací významně neodlišují od těch vypočtených za předpokladu náhodnosti.[2, 3]

Pro pěticí číslic můžeme například využít figury ze známé karetní hry poker. Značí-li symboly a , b , c , d a e libovolné cifry $0, 1, \dots, 9$, dostáváme souhrn variant uvedených v tabulce.[2]

Tab. 1: Poker test – varianty pětic [2]

Varianta	Název	Pravděpodobnost
$a b c d e$	všechny různé	0,302 4
$a a b c d$	jedna dvojka	0,504 0
$a a b b c$	dvě dvojky	0,108 0
$a a a b c$	trojka	0,070 2
$a a a b b$	dvojka a trojka (full house)	0,009 0
$a a a a b$	čtyřka (poker)	0,004 5
$a a a a a$	pětka	0,000 1

K vyhodnocení testu bychom mohli použít například χ^2 test dobré shody. Ten však pro rozumné množství hodnot s výše uvedenými variantami nebude dávat věrohodné výsledky. To je způsobeno velmi nízkou pravděpodobností výskytu některých pětic. V těchto třídách by pak totiž snadno mohla být četnost nižší než 5.

Zmíněný problém a snaha usnadnit programování příslušných výpočtů vedla ke vzniku modifikace tohoto testu, na kterou se nyní blíže podíváme. Je založena na registraci počtu různých číslic v k -ticích. Pro případ, kdy $k = 5$, existují následující kategorie.

Tab. 2: Modifikovaný poker test – varianty pětic [2]

Varianta	Počet různých hodnot
$a b c d e$	5
$a a b c d$	4
$a a a b c, a a b b c$	3
$a a a b b, a a a a b$	2
$a a a a a$	1

Pravděpodobnost jednotlivých variant získáme z následujícího vztahu ze zdroje [2], kde d je počet různých cifer, k označuje délku kombinací číslic a r počet různých hodnot. Složené závorky pak vyjadřují Stirlingovo číslo 2. druhu.[2]

$$p_r = \frac{d(d-1)(d-2)\cdots(d-r+1)}{d^k} \left\{ \begin{matrix} k \\ r \end{matrix} \right\}$$

Použijeme-li dekadickou soustavu a délky jednotlivých k -tic ponecháme 5, tedy $d = 10$ a $k = 5$, získáme následující pravděpodobnosti jednotlivých variant.

Tab. 3: Modifikovaný poker test – varianty pětic s pravděpodobnostmi

Varianta	Počet různých hodnot	Pravděpodobnost
$a b c d e$	5	0,302 4
$a a b c d$	4	0,504 0
$a a a b c, a a b b c$	3	0,180 0
$a a a b b, a a a a b$	2	0,013 5
$a a a a a$	1	0,000 1

Při vykonávání testu je vhodné sloučit poslední dvě kategorie, aby výsledné očekávané četnosti byly dostatečně velké. Samotné číslice získáme obdobně jako u testu výskytu úplných sad číslic, tedy $U_i = [u_i d]$, kde hranaté závorky značí celou část čísla.

1.2.7 Frekvenční test

Frekvenční, nebo také ekvidistribuční test zajišťuje kontrolu, zda jsou jednotlivá čísla v posloupnosti rovnoměrně rozložena po celém intervalu. Postup výpočtu je následující.

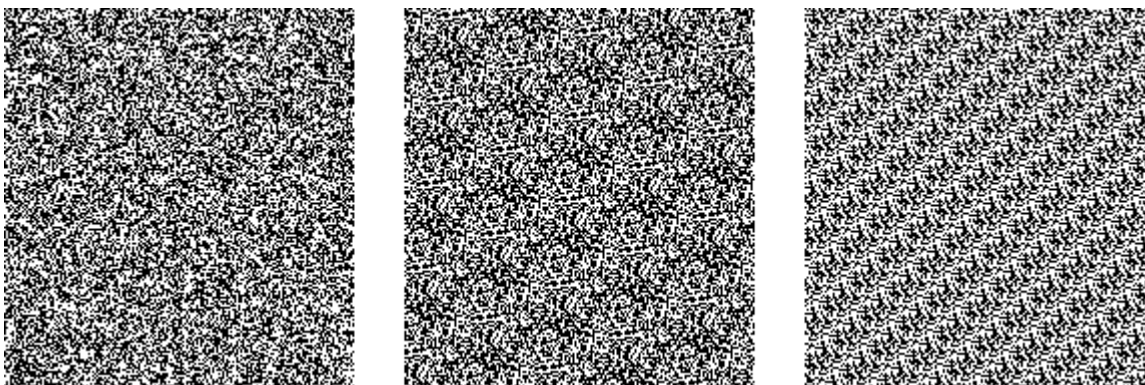
Získáme posloupnost čísel, již chceme ověřit. Tu pomocí χ^2 testu porovnáme s pravděpodobnostmi d libovolně zvolených tříd. Hodnoty pravděpodobností jednotlivých tříd lze pak snadno získat z následujícího vztahu.

$$p_r = \frac{1}{d}$$

1.2.8 Vizuální analýza

Další zajímavou možností, jak zhodnotit kvalitu generátoru či libovolné sekvence čísel, je vizuální analýza. Ta sice nepatří mezi objektivní metody, ale i tak nám může pomoci při prvotním „ohledání“.

Jedná se o velice jednoduchou metodu, kdy jsou do obdélníkové plochy vykreslovány černé a bílé pixely. Černá barva je pak použita v případě, kdy je zkoumané číslo větší jak 0,5. Grafický výstup vizuální analýzy může být následující.



Obr. 3: Vizuální analýza

Na prvním obrázku není patrné žádné opakování či podobnost jednotlivých částí. Černé body se zdají být náhodně rozmístěny, a tak bychom mohli předpokládat, že tento generátor získá dobré hodnocení i v jiných testech.

Druhý výstup už tak optimisticky zhodnotit nemůžeme. Zkušené oko odborníka ihned pozná, že se jednotlivé segmenty opakují. To bude s největší pravděpodobností způsobeno krátkou délkou periody. To znamená, že se generátor dostal velice brzy do smyčky, ve které se opakuje jedna a tatáž posloupnost čísel.

Na posledním zobrazeném testu už i laik pozná, že něco není v pořádku. Je pak zcela jasné, že získané hodnoty rozhodně nesplňují celou řadu kritérií náhodnosti. Generátor této posloupnosti čísel tedy můžeme rovnou zamítnout, aniž bychom ztráceli čas dalšími testy.

Data pro vizuální analýzu byla získána pomocí lineárního kongruenčního generátoru. V každém ze tří případů však byly jinak nastaveny jeho parametry. To nám jasně ukazuje, jak je volba těchto hodnot důležitá a že i dobrý generátor může ve špatných rukou dávat otřesné výsledky.

1.3 Vybraná spojitá rozdělení pravděpodobnosti

Když už víme, co jsou to náhodná čísla, umíme je získat a dovedeme ověřit jejich kvalitu, mohli bychom se pustit rovnou do generování hodnot náhodné proměnné. Předtím si však ještě připomeneme, že existují nějaká rozdělení pravděpodobnosti, kterým tyto hodnoty mohou

odpovídat. Podstatná část implementovaných rozdělení je popsána v již výše zmíněné bakalářské práci [1]. My se zde tedy budeme zabývat pouze těmi nově přidanými.

Než se pustíme na první rozdělení pravděpodobnosti, raději ještě jednou doporučíme oprášení základních znalostí z teorie pravděpodobnosti a matematické statistiky. Všechny potřebné informace čtenář nalezne ve zdroji [1]. Pro podrobnější výklad lze využít například literaturu [9].

1.3.1 Arcsinové rozdělení

Arcsinové rozdělení pravděpodobnosti má parametry a a b , pro které platí $-\infty < a < b < \infty$.

- **Střední hodnota**

$$E(X) = \frac{(a + b)}{2}$$

- **Rozptyl**

$$D(X) = \frac{(b - a)^2}{8}$$

- **Hustota pravděpodobnosti**

$$f(x) = \frac{1}{\pi \sqrt{\frac{x-a}{b-a} \left(1 - \frac{x-a}{b-a}\right)}}, \quad a < x < b$$

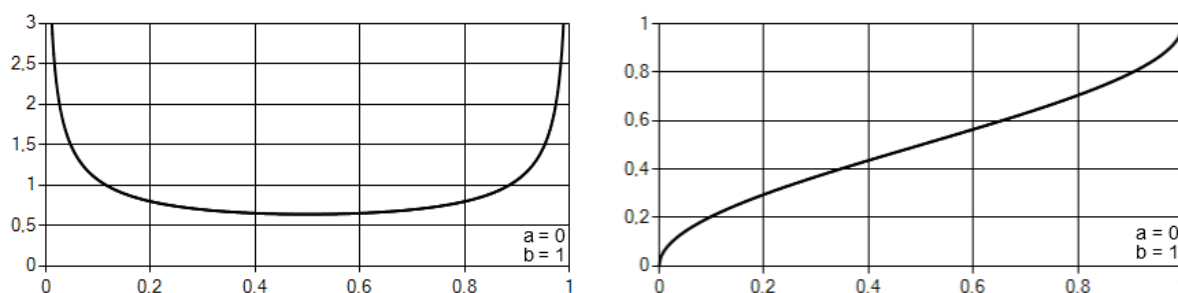
$$= 0, \quad \text{jinak}$$

- **Distribuční funkce**

$$F(x) = 0, \quad x \leq a$$

$$= \frac{2}{\pi} \sin^{-1} \sqrt{\frac{x-a}{b-a}}, \quad a < x < b$$

$$= 1, \quad x \geq b$$



Obr. 4: Hustota pravděpodobnosti a distribuční funkce arcsinova rozdělení

Průběh hustoty pravděpodobnosti připomíná tzv. vanovou křivku, rozdělení by tak bylo možné využít například v teorii spolehlivosti či životnosti.

1.3.2 Rozdělení extrémních hodnot

Rozdělení extrémních hodnot je určeno dvěma parametry. Střední hodnotou μ a rozptylem σ , kde $-\infty < \mu < \infty$ a $0 < \sigma < \infty$.

- **Střední hodnota**

$$E(X) = \mu - \gamma\sigma, \quad \text{kde } \gamma \text{ je Eulerova konstanta } (\gamma \approx 0,57721)$$

- **Rozptyl**

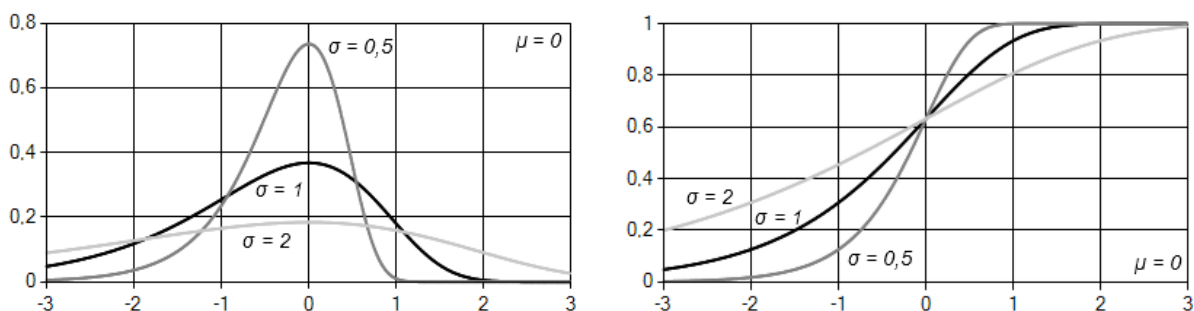
$$D(X) = \frac{\sigma^2 \pi^2}{6}$$

- **Hustota pravděpodobnosti**

$$f(x) = \frac{1}{\sigma} e^{\frac{x-\mu}{\sigma}} e^{-e^{\frac{x-\mu}{\sigma}}}, \quad -\infty < x < \infty$$

- **Distribuční funkce**

$$F(x) = 1 - e^{-e^{\frac{x-\mu}{\sigma}}}, \quad -\infty < x < \infty$$



Obr. 5: Hustota pravděpodobnosti a distribuční funkce rozdělení extrémních hodnot

Teorie extrémních hodnot byla nejčastěji využívána pro modelování rozdělení různých přírodních jevů (dešťové srážky, záplavy, větrné poryvy vzduchu, znečištění vzduchu). Rozdělení výběrového maxima nás může zajímat například při analýze vlivu maximální tíhy sněhu na stavební konstrukce. Při studiu únavy či pevnosti nás může naopak zajímat rozdělení výběrového minima.[10]

Teorie extrémních hodnot tedy vznikla na základě potřeb astronomů, hydrologů a jiných techniků. Začali se o ni zajímat matematici zabývající se teorií pravděpodobnosti a následně i řada statistiků. Dnes tato teorie nachází uplatnění i v ekonomii, kde je využita jako metoda modelování a měření extrémních rizik ve finanční sféře.[10]

1.3.3 Gumbelovo rozdělení pravděpodobnosti

Gumbelovo rozdělení pravděpodobnosti je dáno dvěma parametry. Střední hodnotou μ a rozptylem σ , kde $-\infty < \mu < \infty$ a $0 < \sigma < \infty$.

- **Střední hodnota**

$$E(X) = \mu + \gamma\sigma, \quad \text{kde } \gamma \text{ je Eulerova konstanta } (\gamma \approx 0,57721)$$

- **Rozptyl**

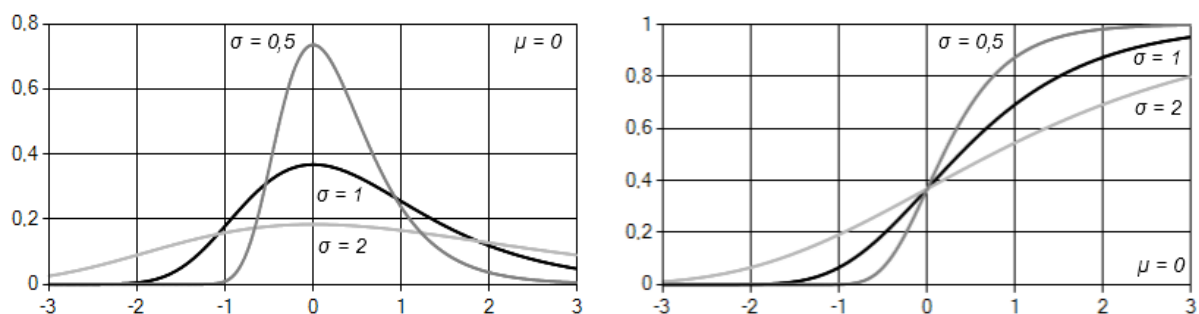
$$D(X) = \frac{\sigma^2 \pi^2}{6}$$

- **Hustota pravděpodobnosti**

$$f(x) = \frac{1}{\sigma} e^{-\frac{x-\mu}{\sigma}} e^{e^{-\frac{x-\mu}{\sigma}}}, \quad -\infty < x < \infty$$

- **Distribuční funkce**

$$F(x) = e^{-e^{-\frac{x-\mu}{\sigma}}}, \quad -\infty < x < \infty$$



Obr. 6: Hustota pravděpodobnosti a distribuční funkce Gumbelova rozdělení

Dle grafů hustoty pravděpodobnosti a distribuční funkce můžeme pozorovat jistou podobnost s předcházejícím rozdělením extrémních hodnot. I jeho využití je obdobné.

1.3.4 Chí-kvadrát rozdělení

Chí-kvadrát rozdělení pravděpodobnosti se v literatuře často označuje symbolem χ_n^2 . Má jeden parametr n , $n \in \mathbb{N}$, jenž nazýváme počet stupňů volnosti.[9] V našem případě je rozdělení doplněno ještě o druhý parametr a , kde $-\infty < a < \infty$, který umožňuje posouvání po ose x .

Toto rozdělení bývá označováno i jako Pearsonovo rozdělení pravděpodobnosti s n stupni volnosti.[9]

- **Střední hodnota**

$$E(X) = a + n$$

- **Rozptyl**

$$D(X) = 2n$$

- **Hustota pravděpodobnosti**

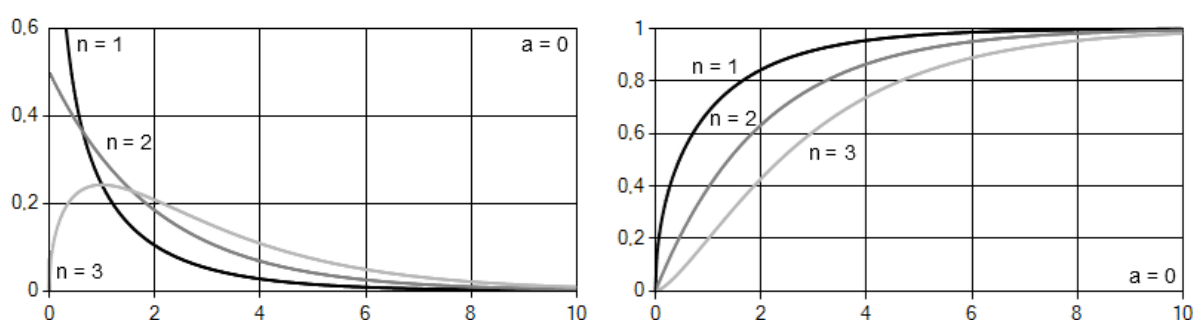
$$f(x) = 0, \quad x \leq a$$

$$= \frac{1}{2^{\frac{n}{2}} \Gamma\left(\frac{n}{2}\right)} (x - a)^{\frac{n-2}{2}} e^{-\frac{x-a}{2}}, \quad x > a$$

- **Distribuční funkce**

$$F(x) = 0, \quad x \leq a$$

$$= \int_0^x \frac{1}{2^{\frac{n}{2}} \Gamma\left(\frac{n}{2}\right)} (x - a)^{\frac{n-2}{2}} e^{-\frac{x-a}{2}} dx, \quad x > a$$



Obr. 7: Hustota pravděpodobnosti a distribuční funkce chí-kvadrát rozdělení

Počet stupňů volnosti určuje počet náhodných veličin v součtu druhých mocnin majících normální normované rozdělení pravděpodobnosti, které se často vyskytují v matematické statistice. Z toho vyplývá i význam tohoto rozdělení.[9]

1.3.5 Kosinové rozdělení

Kosinové rozdělení má parametry a a b , pro které platí $-\infty < a < b < \infty$.

- **Střední hodnota**

$$E(X) = \frac{a + b}{2}$$

- **Rozptyl**

$$D(X) = \frac{(\pi^2 - 8)(b - a)^2}{4\pi^2}$$

- **Hustota pravděpodobnosti**

$$f(x) = \frac{1}{2d} \cos \frac{x - c}{d}, \quad a \leq x \leq b$$

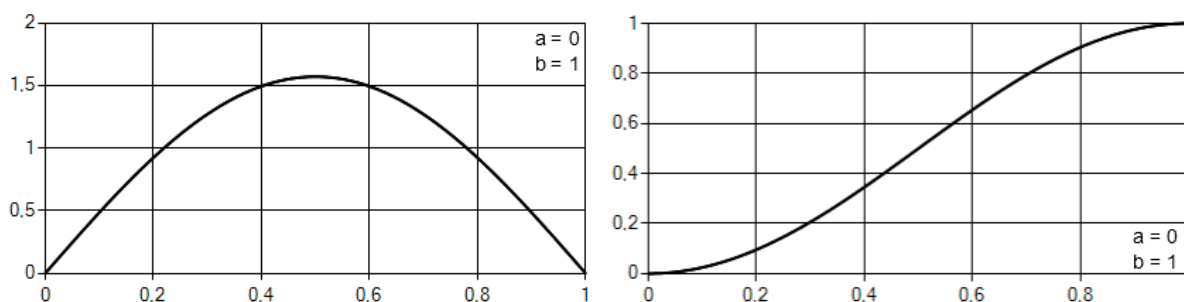
$$= 0, \quad \text{jinak}$$

- **Distribuční funkce**

$$\begin{aligned}
 F(x) &= 0, & x < a \\
 &= \frac{1}{2} \left(1 + \sin \frac{x-a}{b} \right), & a \leq x \leq b \\
 &= 1, & x > b
 \end{aligned}$$

U hustoty pravděpodobnosti a distribuční funkce jsou použity následující substituce:

$$c = \frac{a+b}{2}, d = \frac{b-a}{\pi}.$$



Obr. 8: Hustota pravděpodobnosti a distribuční funkce kosinova rozdělení

Kosinové rozdělení pravděpodobnosti svým tvarem připomíná rozdělení normální, je ovšem omezeno na interval minimální a maximální hodnoty. Mohlo by tak být využito v simulacích právě místo Gaussova rozdělení.

1.3.6 Logaritmické rozdělení

I následující rozdělení má parametry a a b , kde $-\infty < a < b < \infty$, představují minimální a maximální hodnotu.

- **Střední hodnota**

$$E(X) = a + \frac{b-a}{4}$$

- **Rozptyl**

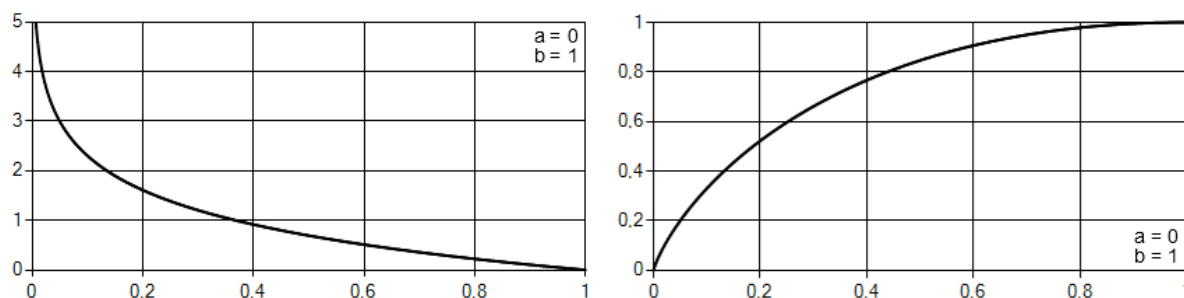
$$D(X) = \frac{7}{144} (b-a)^2$$

- **Hustota pravděpodobnosti**

$$\begin{aligned}
 f(x) &= -\frac{1}{b-a} \ln \frac{x-a}{b-a}, & a < x \leq b \\
 &= 0, & \text{jinak}
 \end{aligned}$$

- **Distribuční funkce**

$$\begin{aligned}
 F(x) &= 0, & x &\leq a \\
 &= \frac{x-a}{b-a} \left(1 - \ln \frac{x-a}{b-a} \right), & a < x &\leq b \\
 &= 1, & x &> b
 \end{aligned}$$



Obr. 9: Hustota pravděpodobnosti a distribuční funkce logaritmického rozdělení

Logaritmické rozdělení pravděpodobnosti se užívá například v hydrologii pro předpověď vlivu srážek na průtok vody v tocích.

1.3.7 Maxwellovo rozdělení

Maxwellovo rozdělení disponuje dvěma parametry. Parametr μ , kde $0 < \mu < \infty$, určuje tvar a parametr posunutí a , kde $-\infty < a < \infty$, který určuje polohu.

- **Střední hodnota**

$$E(X) = a + \mu \sqrt{\frac{8}{\pi}}$$

- **Rozptyl**

$$D(X) = \frac{\mu^2(3\pi - 8)}{\pi}$$

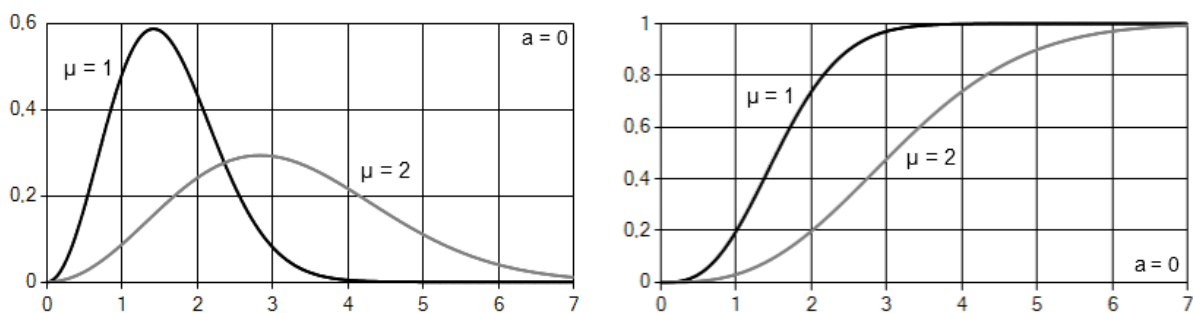
- **Hustota pravděpodobnosti**

$$\begin{aligned}
 f(x) &= \sqrt{\frac{2}{\pi}} \frac{(x-a)^2 e^{-\frac{(x-a)^2}{2\mu^2}}}{\mu^3}, & x &> 0 \\
 &= 0, & x &\leq 0
 \end{aligned}$$

- **Distribuční funkce**

$$\begin{aligned}
 F(x) &= 0, & x &\leq a \\
 &= \Gamma\left(\frac{1}{2}, \frac{(x-a)^2}{2\mu^2}\right), & x &> a
 \end{aligned}$$

$\Gamma(p, x)$ je neúplná gama funkce.



Obr. 10: Hustota pravděpodobnosti a distribuční funkce Maxwellova rozdělení

Toto rozdělení pravděpodobnosti nachází využití například v kinetické teorii plynů nebo kvantové fyzice.[9]

1.3.8 Paretovo rozdělení

Paretovo rozdělení má tři parametry a, b, m , kde $0 < a, b < \infty$ a $-\infty < m < \infty$. [9]

- **Střední hodnota**

$$E(X) = m + \frac{ba}{a-1}, \quad a > 1$$

- **Rozptyl**

$$D(X) = \frac{ab^2}{(a-1)^2(a-2)}, \quad a > 2$$

- **Hustota pravděpodobnosti**

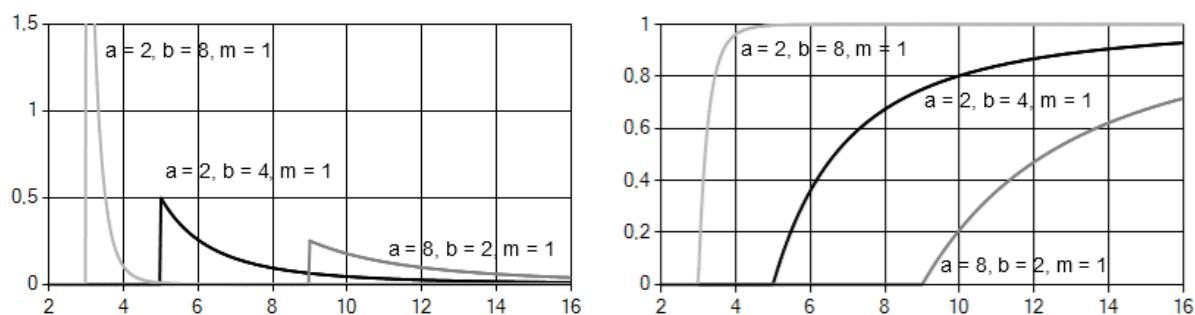
$$f(x) = \frac{ab^a}{(x-m)^{a+1}}, \quad x > m+b$$

$$= 0, \quad x \leq m+b$$

- **Distribuční funkce**

$$F(x) = 1 - \left(\frac{b}{x-m}\right)^a, \quad x > m+b$$

$$= 0, \quad x \leq m+b$$



Obr. 11: Hustota pravděpodobnosti a distribuční funkce Paretova rozdělení

„Paretovo rozdělení se používá jako rozdělení pojistných plnění zejména při modelování jejich extrémních hodnot. Toto rozdělení odstraňuje některé nedostatky, které se objeví při použití exponenciálního rozdělení, neboť konverguje k nule pomaleji než exponenciální rozdělení. Je využíváno také při modelování příjmů obyvatelstva.“[9]

1.4 Vybraná diskrétní rozdělení pravděpodobnosti

Nyní se dostáváme k diskrétním rozdělením pravděpodobnosti, která se od spojitých v některých ohledech značně liší. Základní rozdíly čtenář nalezne ve zdroji [1], podrobnější rozbor pak například ve zdroji [9].

1.4.1 Bernoulliho rozdělení

Toto rozdělení, někdy označováno také jako alternativní, má pouze jeden parametr p , kde $p \in \langle 0; 1 \rangle$, určuje pravděpodobnost úspěchu. Náhodná veličina nabývá hodnot 0 a 1.

- **Střední hodnota**

$$E(X) = p$$

- **Rozptyl**

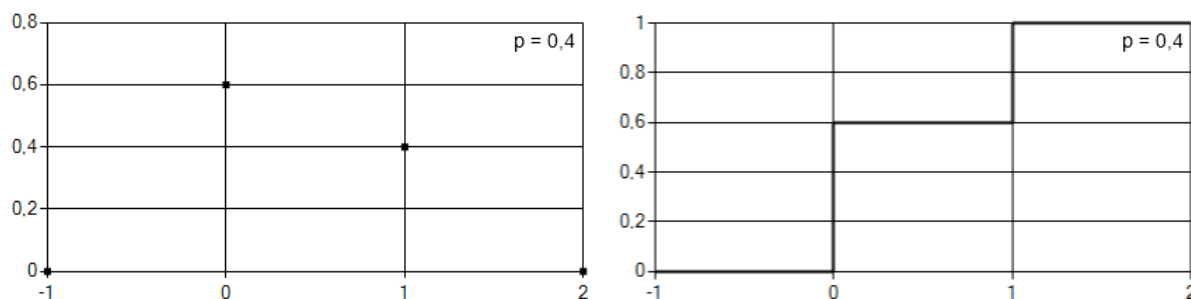
$$D(X) = p(1 - p)$$

- **Pravděpodobnostní funkce**

$$\begin{aligned} p(x) &= p, & x &= 1 \\ &= 1 - p, & x &= 0 \end{aligned}$$

- **Distribuční funkce**

$$\begin{aligned} F(x) &= 0, & x &< 0 \\ &= 1 - p, & 0 &\leq x < 1 \\ &= 1, & x &\geq 1 \end{aligned}$$



Obr. 12: Pravděpodobnostní funkce a distribuční funkce Bernoulliho rozdělení

Toto rozdělení můžeme využít kdekoliv, kde mohou nastat pouze dva stavy, přičemž pravděpodobnost jednoho ze stavů je dána parametrem p . Jako příklad může sloužit hod mincí.

1.4.2 Binomické rozdělení

Náhodná veličina binomického rozdělení s dvěma parametry p a n , kde $p \in \langle 0; 1 \rangle$ a $n \in \mathbb{N}$, nabývá hodnot $0, 1, 2, \dots, n$. Pro parametr $n = 1$ přechází toto rozdělení v Bernoulliho s parametrem p .

- **Střední hodnota**

$$E(X) = np$$

- **Rozptyl**

$$D(X) = np(1 - p)$$

- **Pravděpodobnostní funkce**

$$p(x) = \binom{n}{x} p^x (1 - p)^{n-x}, \quad x \in \{0, 1, \dots, n\}$$

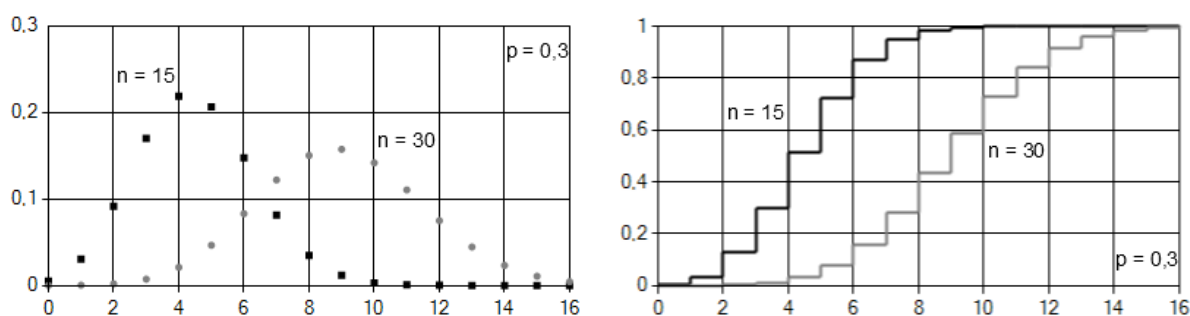
$$= 0, \quad \text{jinak}$$

- **Distribuční funkce**

$$F(x) = 0, \quad x < 0$$

$$= \sum_{i=0}^x \binom{n}{i} p^i (1 - p)^{n-i}, \quad 0 \leq x \leq n$$

$$= 1, \quad x > n$$



Obr. 13: Pravděpodobnostní funkce a distribuční funkce binomického rozdělení

Tímto rozdělením se řídí například nezávislý výběr, někdy také nazývaný výběr s vrácením.[9]

1.4.3 Geometrické rozdělení

Geometrické rozdělení s parametrem p , kde $p \in \langle 0; 1 \rangle$, reprezentuje pravděpodobnost x neúspěšných Bernoulliho pokusů před prvním úspěšným.

- **Střední hodnota**

$$E(X) = \frac{(1 - p)}{p}$$

- Rozptyl

$$D(X) = \frac{(1-p)}{p^2}$$

- Pravděpodobnostní funkce

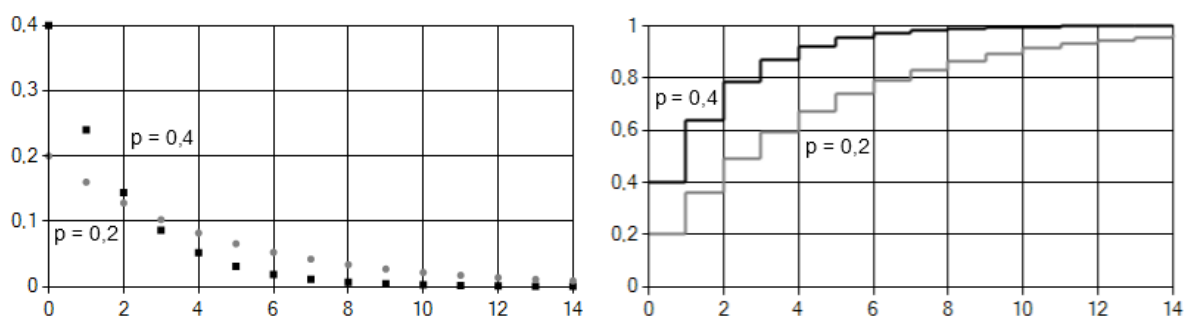
$$p(x) = p(1-p)^x, \quad x \in \{0, 1, \dots, n\}$$

$$= 0, \quad \text{jinak}$$

- Distribuční funkce

$$F(x) = 1 - (1-p)^{x+1}, \quad x \geq 0$$

$$= 0, \quad \text{jinak}$$



Obr. 14: Pravděpodobnostní funkce a distribuční funkce geometrického rozdělení

Toto rozdělení popisuje pravděpodobnost počtu pokusů do prvního úspěchu. Představme si například, že se nám porouchal nějaký spotřebič a my se zkusíme dovolat na linku technické podpory. Dle tohoto rozdělení můžeme předpovědět pravděpodobnost, že se dovoláme na první, druhý nebo některý další pokus.

1.4.4 Hypergeometrické rozdělení

Hypergeometrické rozdělení má tři parametry N , M a n , kde $N, M, n \in \mathbb{N}$ a $N \geq M, n$. Náhodná veličina X nabývá hodnot x , kde platí $\max\{0, M - N + n\} \leq x \leq \min\{M, n\}$. [9]

- Střední hodnota

$$E(X) = n \frac{M}{N}$$

- Rozptyl

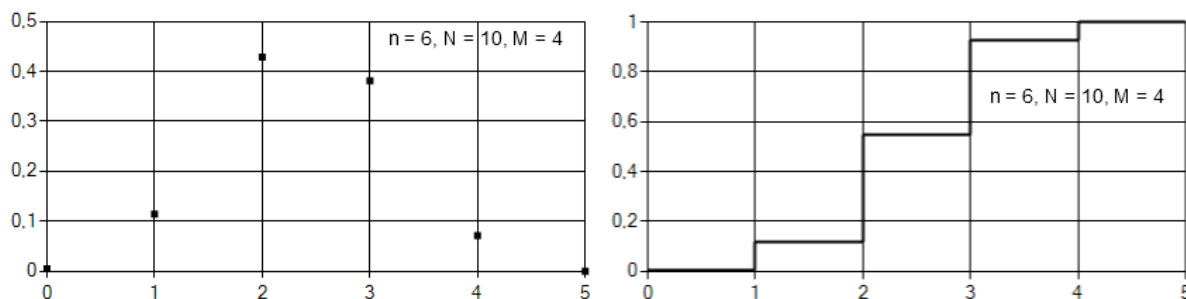
$$D(X) = n \frac{M}{N} \left(1 - \frac{M}{N}\right) \frac{N-n}{N-1}$$

- Pravděpodobnostní funkce

$$p(x) = \frac{\binom{M}{x} \binom{N-M}{n-x}}{\binom{N}{n}}$$

- **Distribuční funkce**

$$F(x) = \sum_{i=0}^x \frac{\binom{M}{i} \binom{N-M}{n-i}}{\binom{N}{n}}$$



Obr. 15: Pravděpodobnostní funkce a distribuční funkce hypergeometrického rozdělení

Podle tohoto rozdělení se řídí například tzv. nezávislý výběr. Tedy, v souboru N prvků má M prvků jistou vlastnost. Vybíráme n prvků, a to buď postupně bez vracení, nebo všechny najednou. Získáme pravděpodobnost, že právě x vybraných prvků bude mít požadovanou vlastnost.[9]

1.4.5 Negativně binomické rozdělení

Toto rozdělení má dva parametry n a p , kde $n > 0$ a $p \in (0; 1)$.

- **Střední hodnota**

$$E(X) = \frac{n(1-p)}{p}$$

- **Rozptyl**

$$D(X) = \frac{n(1-p)}{p^2}$$

- **Pravděpodobnostní funkce**

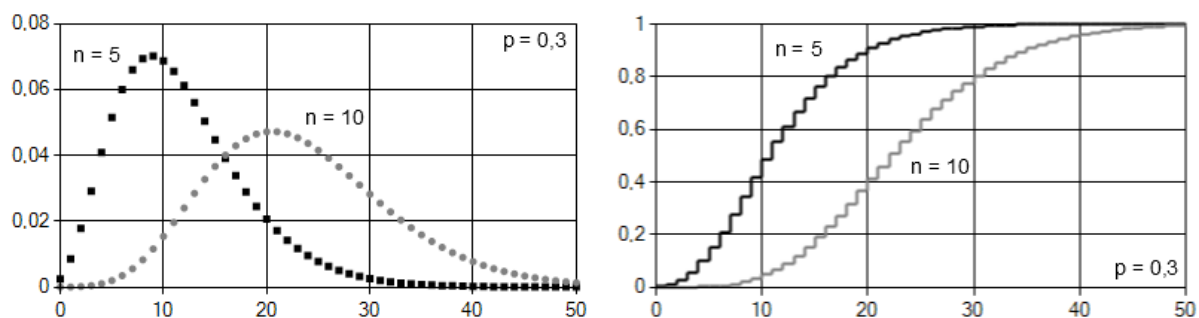
$$p(x) = \frac{(n+x-1)!}{x!(n-1)!} p^n (1-p)^x, \quad x \in \{0, 1, \dots, n\}$$

$$= 0, \quad \text{jinak}$$

- **Distribuční funkce**

$$F(x) = \sum_{i=0}^x \frac{(n+i-1)!}{i!(n-1)!} p^n (1-p)^i, \quad x \geq 0$$

$$= 0, \quad \text{jinak}$$

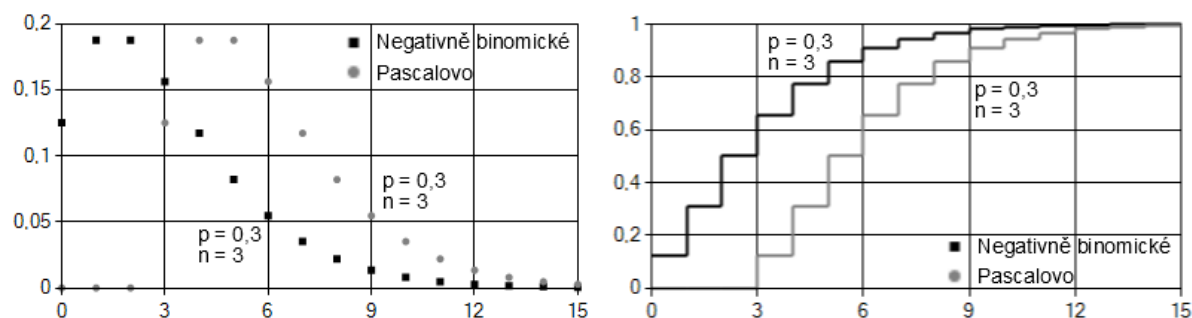


Obr. 16: Pravděpodobnostní funkce a distribuční funkce negativně binomického rozdělení

Negativně binomické rozdělení v posloupnosti opakovaných nezávislých Bernoulliho pokusů představuje, kolik předchází n -tému pokusu neúspěchů x . V pojišťovnictví se tímto rozdělením řídí například počet pojistných plnění v případě nehomogenního rizika.[9]

1.4.6 Pascalovo rozdělení

V některých publikacích bývá Pascalovo rozdělení označováno jako pouze jiný název pro rozdělení negativně binomické. Má také dva parametry n a p , kde $n > 0$ a $p \in \langle 0; 1 \rangle$. V našem případě se však liší, což ukazuje následující graf pravděpodobnostní a distribuční funkce.



Obr. 17: Rozdíl mezi negativně binomickým a Pascalovým rozdělením

- **Střední hodnota**

$$E(X) = \frac{n}{p}$$

- **Rozptyl**

$$D(X) = \frac{n(1-p)}{p^2}$$

- **Pravděpodobnostní funkce**

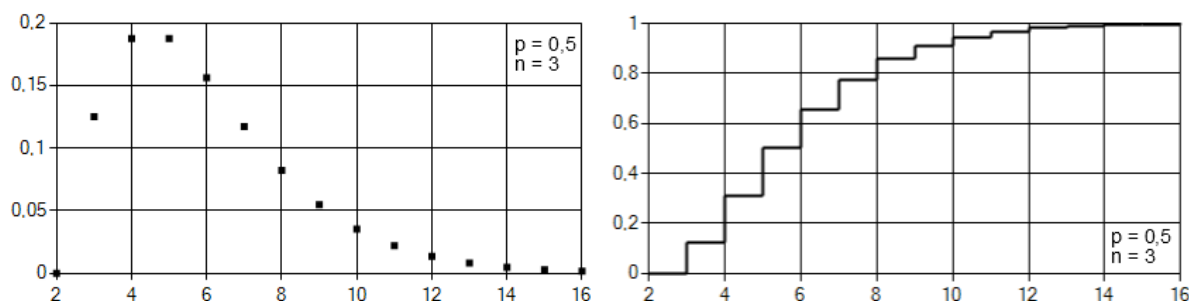
$$p(x) = \frac{(x-1)!}{(x-n)!(n-1)!} p^n (1-p)^{x-n}, \quad x \in \{n, n+1, \dots\}$$

$$= 0, \quad \text{jinak}$$

- **Distribuční funkce**

$$F(x) = \sum_{i=0}^x \frac{(i-1)!}{(i-n)!(n-1)!} p^n (1-p)^{i-n}, \quad x \geq n$$

$$= 0, \quad \text{jinak}$$



Obr. 18: Pravděpodobnostní funkce a distribuční funkce Pascalova rozdělení

Vzhledem k podobnosti s negativně binomickým rozdělením se příliš neliší ani jejich využití. Rozdíl je v tom, že u Pascalova rozdělení dochází k posunutí na ose x . To je určeno hodnotou parametru n , což je krásně vidět na výše uvedených grafech porovnávajících pravděpodobnostní a distribuční funkce obou rozdělení.

1.4.7 Poissonovo rozdělení

Poissonovo rozdělení má pouze jeden parametr λ , pro který platí $\lambda > 0$.

- **Střední hodnota**

$$E(X) = \lambda$$

- **Rozptyl**

$$D(X) = \lambda$$

- **Pravděpodobnostní funkce**

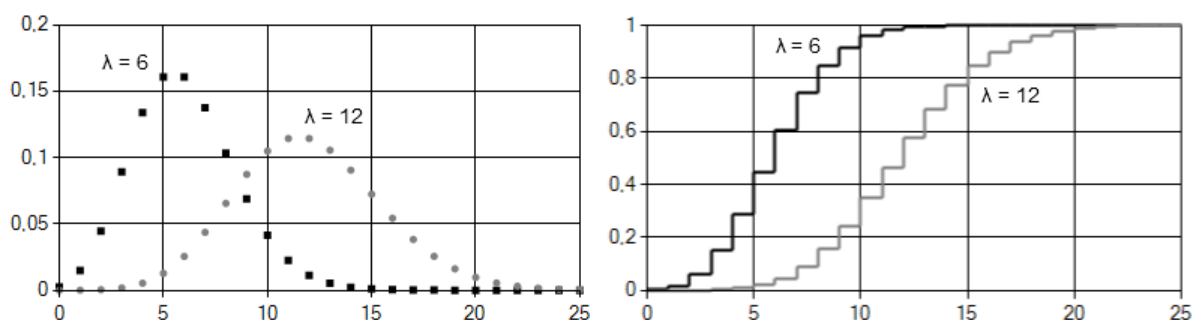
$$p(x) = e^{-\lambda} \frac{\lambda^x}{x!}, \quad x \in \{0, 1, \dots\}$$

$$= 0, \quad \text{jinak}$$

- **Distribuční funkce**

$$F(x) = \sum_{i=0}^x e^{-\lambda} \frac{\lambda^i}{i!}, \quad x \geq 0$$

$$= 0, \quad \text{jinak}$$



Obr. 19: Pravděpodobnostní funkce a distribuční funkce Poissonova rozdělení

Podle tohoto rozdělení se často řídí počty nějakých událostí za časovou jednotku. Mohou to být například počty zákazníků přicházejících do systému obsluhy či počty poruch zařízení.

1.4.8 Rovnoměrné rozdělení

Diskrétní rovnoměrné rozdělení má, obdobně jako jeho spojitá varianta, dva parametry a a b představující minimální a maximální hodnotu, kde $a, b \in \mathbb{Z}$ a $-\infty < a < b < \infty$.

- **Střední hodnota**

$$E(X) = \frac{a + b}{2}$$

- **Rozptyl**

$$D(X) = \frac{(b - a + 1)^2 - 1}{12}$$

- **Pravděpodobnostní funkce**

$$p(x) = \frac{1}{b - a + 1}, \quad x \in \{a, \dots, b\}$$

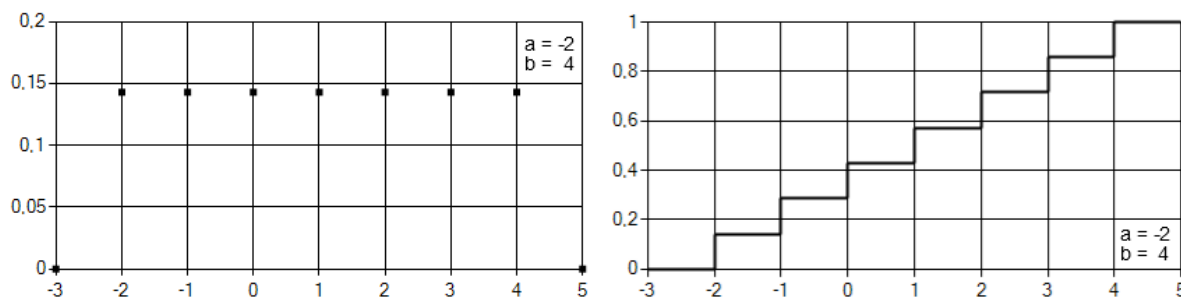
$$= 0, \quad \text{jinak}$$

- **Distribuční funkce**

$$F(x) = 0, \quad x < a$$

$$= \frac{x - a + 1}{b - a + 1}, \quad a \leq x < b$$

$$= 1, \quad x \geq b$$



Obr. 20: Pravděpodobnostní funkce a distribuční funkce rovnoměrného rozdělení

Učebnicovým příkladem rovnoměrného rozdělení je například hod hrací kostkou.

1.5 Transformace náhodných čísel

Když už známe všechny potřebné informace, můžeme se pustit do generování hodnot náhodné proměnné. To probíhá ve dvou krocích. Nejdříve musíme získat náhodné číslo, tedy hodnotu s rovnoměrným rozdělením pravděpodobnosti na intervalu $(0; 1)$. Následně toto číslo transformujeme tak, aby odpovídalo zvolenému rozdělení pravděpodobnosti s požadovanými parametry.

V této kapitole si uvedeme tři nejčastěji používané metody. Ty jsou velice stručně popsány i v publikaci [1], my si je zde však probereme podrobněji. Na konci kapitoly pak zmíníme další využívané principy transformace.

1.5.1 Metoda inverzní transformace

Předpokládejme, že máme náhodnou veličinu, pro jejíž hustotu pravděpodobnosti platí následující předpis.[2]

$$f(x) > 0, \quad a < x < b$$

$$f(x) = 0, \quad \text{jinak}$$

Pak je distribuční funkce $F(x)$ v rozmezí $(a; b)$ rostoucí a tento interval zobrazuje na $(0; 1)$. Existuje tedy vzájemně jednoznačné přiřazení mezi hodnotami $x \in (a; b)$ a $u \in (0; 1)$. Díky tomu můžeme zvolené číslo u převést na hodnotu x následovně.[2]

$$u = F(x)$$

Nyní musíme nalézt inverzní distribuční funkci F^{-1} , tím vyjádříme x . Pokud existuje F^{-1} v explicitní tvaru, získáme předpis, pomocí kterého je možná zmíněnou hodnotu x získat.[2] Pokud jsou splněny všechny uvedené předpoklady, nic nám nebrání v generování hodnot náhodné proměnné následujícím způsobem.

$$x = F^{-1}(u), \quad u \in (0; 1)$$

Samotnou transformaci si ukážeme na jednom z příkladů, který bylo nutné vypočítat při vývoji přiložené aplikace. Konkrétně se jedná o Paretovo rozdělení pravděpodobnosti s následující hustotou pravděpodobnosti a distribuční funkcí. Význam jednotlivých parametrů a další informace o tomto rozdělení je možné nalézt v části s vybranými spojitými rozděleními, konkrétně v kapitole 1.3.8.

$$\begin{aligned} f(x) &= \frac{ab^a}{(x-m)^{a+1}}, & x > m+b \\ &= 0, & x \leq m+b \\ F(x) &= 1 - \left(\frac{b}{x-m}\right)^a, & x > m+b \\ &= 0, & x \leq m+b \end{aligned}$$

Nyní nalezneme inverzní distribuční funkci, díky které je možné generovat pseudonáhodná čísla s Paretovým rozdělením pravděpodobnosti.

$$x = F^{-1}(u) = m + \frac{b}{\sqrt[a]{1-u}}, \quad 0 < u < 1$$

1.5.2 Zamítací metoda

Dalším možným způsobem transformace je zamítací metoda. Ta bývá označována i jako vylučovací a předpokládá následující: hustota pravděpodobnosti f je ohraničena uzavřeným intervalem $\langle a; b \rangle$. Mimo tento interval je $f(x) = 0$ a existuje takové číslo c , že $f(x) \leq c$ pro $x \in \langle a; b \rangle$. [2]

Princip metody spočívá v generování bodů $[x; y]$ rovnoměrně rozložených v obdélníku, v němž je uzavřen graf $f(x)$. Pokud $y \leq f(x)$, považujeme x za vygenerovanou hodnotu. [2]

Algoritmus transformace je následující. Vygenerujeme dvojici náhodných čísel u_1 a u_2 . Z těchto hodnot pomocí vzorců získáme požadovaný bod: $x = a + (b-a)u_1$ a $y = cu_2$. Je zřejmé, že jsou x na $\langle a; b \rangle$ a y na $\langle 0; c \rangle$ rovnoměrně rozloženy. Jestliže je $y > f(x)$, vrátíme se zpět na začátek algoritmu, v opačném případě je x vygenerovaná hodnota. [2]

Efektivita této metody se někdy charakterizuje koeficientem využití, který je dán poměrem získaných hodnot x k celkovému počtu pokusů, tedy počtu vygenerovaných dvojic náhodných čísel. Pokud je koeficient nějaké funkce f nízký, znamená to, že na získání jedné hodnoty x je potřeba značného počtu hodnot u_1 a u_2 . [2]

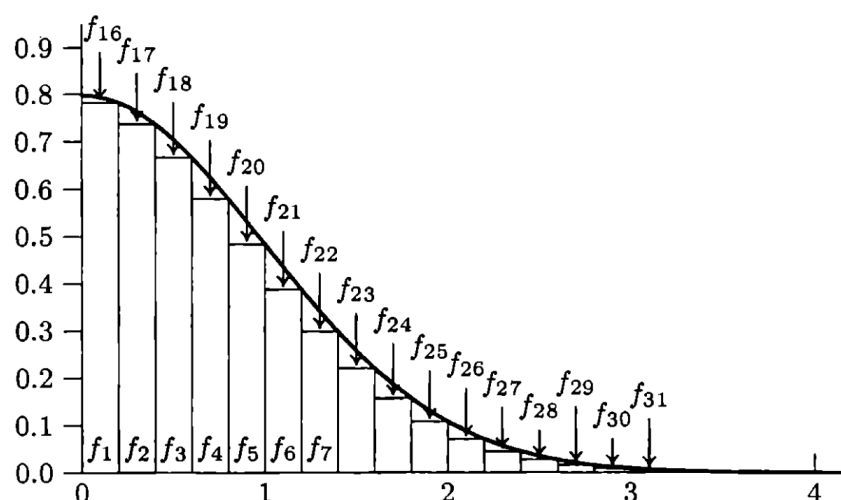
1.5.3 Kompoziční metoda

Podstatou této metody je vyjádření hustoty pravděpodobnosti náhodné veličiny v uvedeném tvaru.[2]

$$f(x) = \sum_{i=1}^n p_i f_i(x)$$

Víme, že celková plocha pod grafem je rovna jedné a každé $f_i(x) \geq 0$. Výše uvedený předpis pak můžeme pohodlně vyjádřit graficky. Plochu pod $f(x)$ rozdělíme do n částí, přičemž část odpovídající $f_i(x)$ má plochu p_i . To ilustruje níže uvedený obrázek 21. Zachycuje situaci, kdy je plocha rozdělena na 31 částí – v uvedeném pořadí 15 obdélníků, 15 klínků a zbývající část pro $x \geq 3$, tzv. okraj. Dle uvedené dekompozice bývá tento způsob transformace označován i jako metoda obdélník-klín-okraj. Hustota pravděpodobnosti uvedeného příkladu má následující tvar.[3]

$$f(x) = \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2}}$$



Obr. 21: Princip kompoziční metody [3]

Většina různých rozdělení se dá tímto způsobem zvládnout velice snadno, protože se stanou triviálními modifikacemi rovnoměrného rozdělení. Výsledná metoda je pak velice efektivní, průměrná doba provádění je totiž velmi krátká. Například v uvedeném příkladu použijeme rovnoměrné rozdělení přibližně v 92% případů.[3]

1.5.4 Další možnosti transformace

V některých případech můžeme využít jiných principů a vlastností náhodných veličin. Pokud jsou například náhodné veličiny X_1 a X_2 nezávislé, platí pro jejich distribuční funkce F_1

a F_2 následující vztahy: $\min(X_1; X_2)$ má distribuční funkci $F_1(x)F_2(x)$ a $\max(X_1; X_2)$ pak $F_1(x) + F_2(x) - F_1(x)F_2(x)$. [2]

Další zajímavou možností transformace je **polární metoda pro normální veličiny**. Už dle názvu můžeme usoudit, že se bude jednat o generátor hodnot normálního rozdělení pravděpodobnosti. Jednotlivé části algoritmu si nyní popíšeme.

V prvním kroku vygenerujeme dvě nezávislá čísla u_1 a u_2 s rovnoměrným rozdělením pravděpodobnosti na intervalu 0 až 1. Ta přiřadíme do v_1 a v_2 tak, aby náležela intervalu od -1 do 1 . K tomu můžeme použít následující jednoduchý předpis: $v = 2u - 1$. [3]

V druhém kroku zjistíme, zda platí $V < 1$, kde $V = v_1^2 + v_2^2$. Pokud ano, pokračujeme dále, v opačném případě se vrátíme na začátek algoritmu. První dva kroky se provedou v průměru 1,27krát se standardní odchylkou 0,59. [3]

Nyní můžeme dopočítat výslednou hodnotu. Tu si oproti literatuře [3] upravíme pro normální rozdělení pravděpodobnosti s libovolnými parametry. Předpis pro její výpočet je následující.

$$x = \mu + \sigma v_1 \sqrt{\frac{-2 \ln V}{V}}, \quad V \neq 0$$

$$= \mu, \quad V = 0$$

Při hlubším zájmu o problematiku transformace náhodných čísel doporučujeme prostudovat odpovídající kapitoly ve zdrojích [3] a [6].

1.6 Speciální funkce

Poslední teoretická kapitola se zabývá doplňkovými funkcemi, které byly potřeba například při výpočtech hustot pravděpodobnosti či pravděpodobnostních a distribučních funkcí jednotlivých rozdělení, případně při testování náhodnosti. Byla tak přidána i možnost jejich samostatného výpočtu, a proto se o nich stručně zmíníme i zde.

1.6.1 Faktoriál

Faktoriál n představuje součin všech kladných celých čísel, která jsou menší nebo rovna n a je označen pomocí vykřičníku. Definován je pro $n \in \mathbb{Z}_0^+$ následovně.

$$n! = \prod_{k=1}^n k, \quad n > 0$$

$$= 1, \quad n = 0$$

1.6.2 Kombinační číslo

Kombinace k -té třídy z n prvků jsou všechny možné varianty výběru k různých prvků z množiny n prvků. V tomto případě nezáleží na pořadí uvnitř jednotlivých kombinací.[11] Kombinační číslo je pak definováno pro $n, k \in \mathbb{Z}_0^+$ následovně.

$$\binom{n}{k} = \frac{n!}{k! (n-k)!}, \quad n \geq k \geq 0$$

1.6.3 Stirlingovo číslo druhého druhu

Stirlingovo číslo druhého druhu obecně udává počet všech možností, kterými lze rozdělit n prvků do k disjunktních neprázdných podmnožin. Definováno je pro $n, k \in \mathbb{Z}_0^+$. Platí pro ně následující vlastnosti.[2]

$$\begin{aligned} \left\{ \begin{matrix} n \\ 0 \end{matrix} \right\} &= 0, & \left\{ \begin{matrix} n \\ 1 \end{matrix} \right\} &= 1, & \left\{ \begin{matrix} n \\ 2 \end{matrix} \right\} &= 2^{n-1} - 1 \\ \left\{ \begin{matrix} 0 \\ 0 \end{matrix} \right\} &= 1, & \left\{ \begin{matrix} 0 \\ 1 \end{matrix} \right\} &= 1, & \left\{ \begin{matrix} n \\ n \end{matrix} \right\} &= 1 \end{aligned}$$

Pomocí těchto vlastností můžeme Stirlingovo číslo 2. druhu získat na základě rekurentního předpisu ze zdroje [2].

$$\left\{ \begin{matrix} n \\ k \end{matrix} \right\} = k \left\{ \begin{matrix} n-1 \\ k \end{matrix} \right\} + \left\{ \begin{matrix} n-1 \\ k-1 \end{matrix} \right\}$$

Explicitní rovnice pro výpočet lze nalézt ve zdroji [12] a je následující.

$$\left\{ \begin{matrix} n \\ k \end{matrix} \right\} = \frac{1}{k!} \sum_{i=0}^k (-1)^{k-i} \binom{k}{i} i^n$$

1.6.4 Chybová funkce

Jedná se o speciální funkci, která najde své využití především v matematické statistice, teorii pravděpodobnosti a fyzice (vedení tepla, difúze). Můžeme s její pomocí například odhadnout hodnotu distribuční funkce normálního rozdělení pravděpodobnosti. Dle zdroje [13] je definována následovně.

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt, \quad -\infty < x < \infty$$

Tato i následující funkce jsou již velice složité. Navíc k nim neexistuje ani funkce primitivní, výsledek integrace je tak nutné získat pomocí numerických metod a je pouze tabelován. Při hlubším zájmu tak doporučujeme prostudovat odbornou literaturu.

1.6.5 Gama funkce

Gama funkce nalézá uplatnění v podobných oborech jako funkce chybová. Konkrétně je pak využívána při výpočtech týkajících se gama rozdělení pravděpodobnosti. Dle zdroje [13] je definována následovně.

$$\Gamma(p) = \int_0^{\infty} t^{p-1} e^{-t} dt, \quad p > 0$$

Pokud $p \in \mathbb{N}$, můžeme gama funkci použít k výpočtu faktoriálu.[13]

$$p! = \Gamma(p + 1)$$

Nakonec ještě zmíníme užitečné vlastnosti této funkce převzaté ze zdroje [9].

$$\Gamma(p + 1) = p\Gamma(p), \quad \Gamma(1) = 1, \quad \Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}$$

1.6.6 Beta funkce

Beta funkci využijeme při výpočtech týkajících se beta rozdělení pravděpodobnosti. Zdroj [13] udává její definici takto.

$$B(a, b) = \int_0^1 t^{a-1} (1 - t)^{b-1} dt$$

Hodnotu beta funkce můžeme získat i pomocí gama funkcí, jak je zmíněno ve zdroji [9].

$$B(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a + b)}, \quad a, b > 0$$

1.6.7 Neúplná gama a beta funkce

Tyto funkce vycházejí z „úplných“ gama a beta funkcí, jsou ovšem doplněny o další parametr x . Jejich definice jsou dle zdroje [13] následující.

$$\Gamma(p, x) \equiv \frac{1}{\Gamma(p)} \int_0^x e^{-t} t^{p-1} dt, \quad p > 0, x \geq 0$$

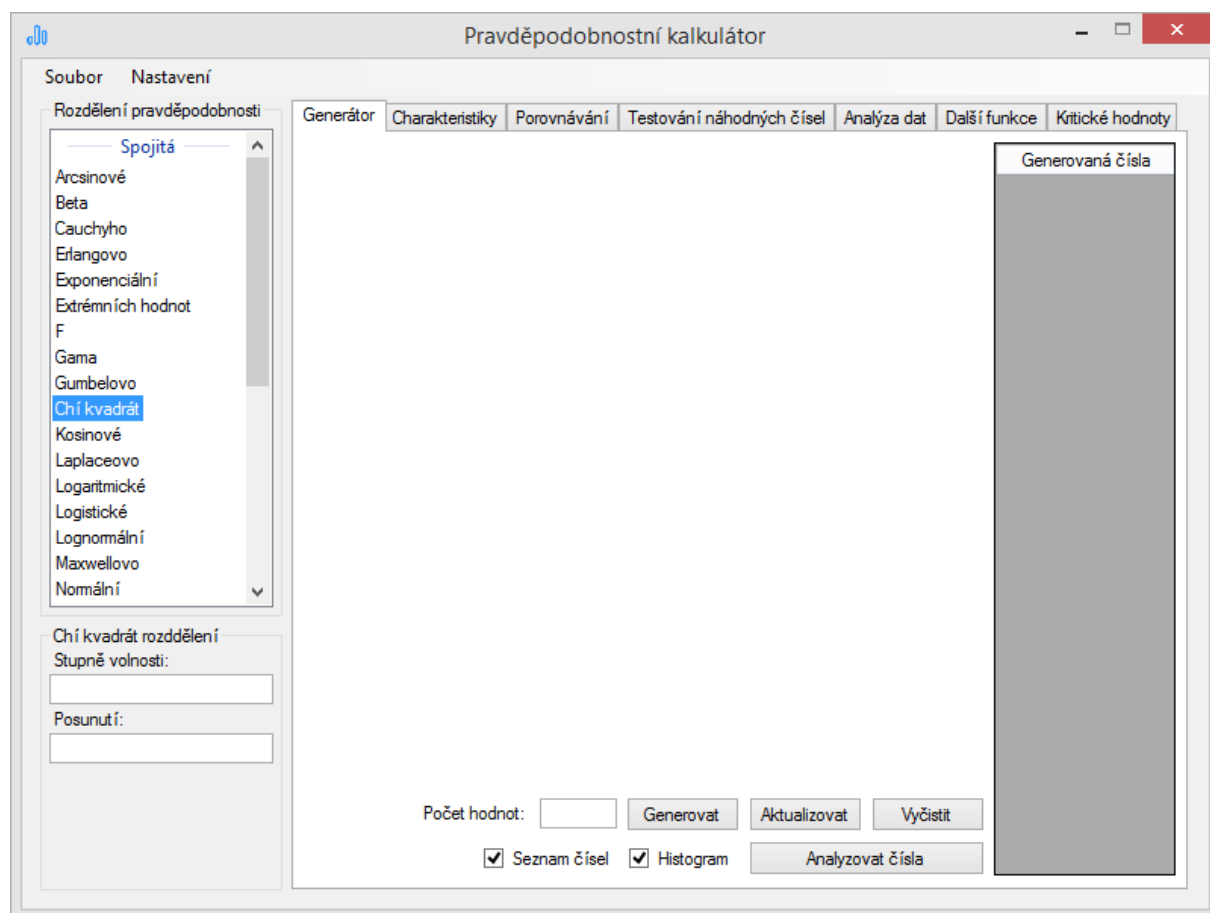
$$B(a, b, x) \equiv \frac{1}{B(a, b)} \int_0^x t^{a-1} (1 - t)^{b-1} dt, \quad a, b > 0, x \in \langle 0; 1 \rangle$$

2 OVLÁDÁNÍ APLIKACE

V této části se dostáváme k samotné aplikaci. Řekneme si, jaké funkcionality nabízí, jak je co nejsnáze využít a jak dosáhnout co možná nejlepších výsledků. Podrobně si popíšeme ovládání aplikace i strukturu jednotlivých formulářů. K tomu dobře poslouží i vložené ilustrace. Po přečtení této kapitoly dokáže čtenář přiložený nástroj efektivně využívat, a tak v krátké době a bez většího úsilí získá kredibilní údaje a potřebné informace.

Nejdříve se podíváme na okno aplikace jako celek. To je složeno z hlavního menu a dvou funkčních částí. V menu je možné ukončit program, uložit vygenerovaná čísla a změnit jednotlivá nastavení. K tomu všemu se však dostaneme až později.

První stěžejní část pak slouží k výběru požadovaného rozdělení pravděpodobnosti a nastavení jeho parametrů. Tento segment je společný pro všechny funkce, kde je nutné rozdělení pravděpodobnosti zvolit. Po vybrání ze seznamu jsou zobrazena odpovídající zadávací pole, která jsou rozlišena názvy parametrů. K dispozici jsou kromě běžně používaných i některá méně obvyklá rozdělení pravděpodobnosti, která nelze nalézt v běžně dostupných softwarech. Celkem lze vybírat z 31 možností.

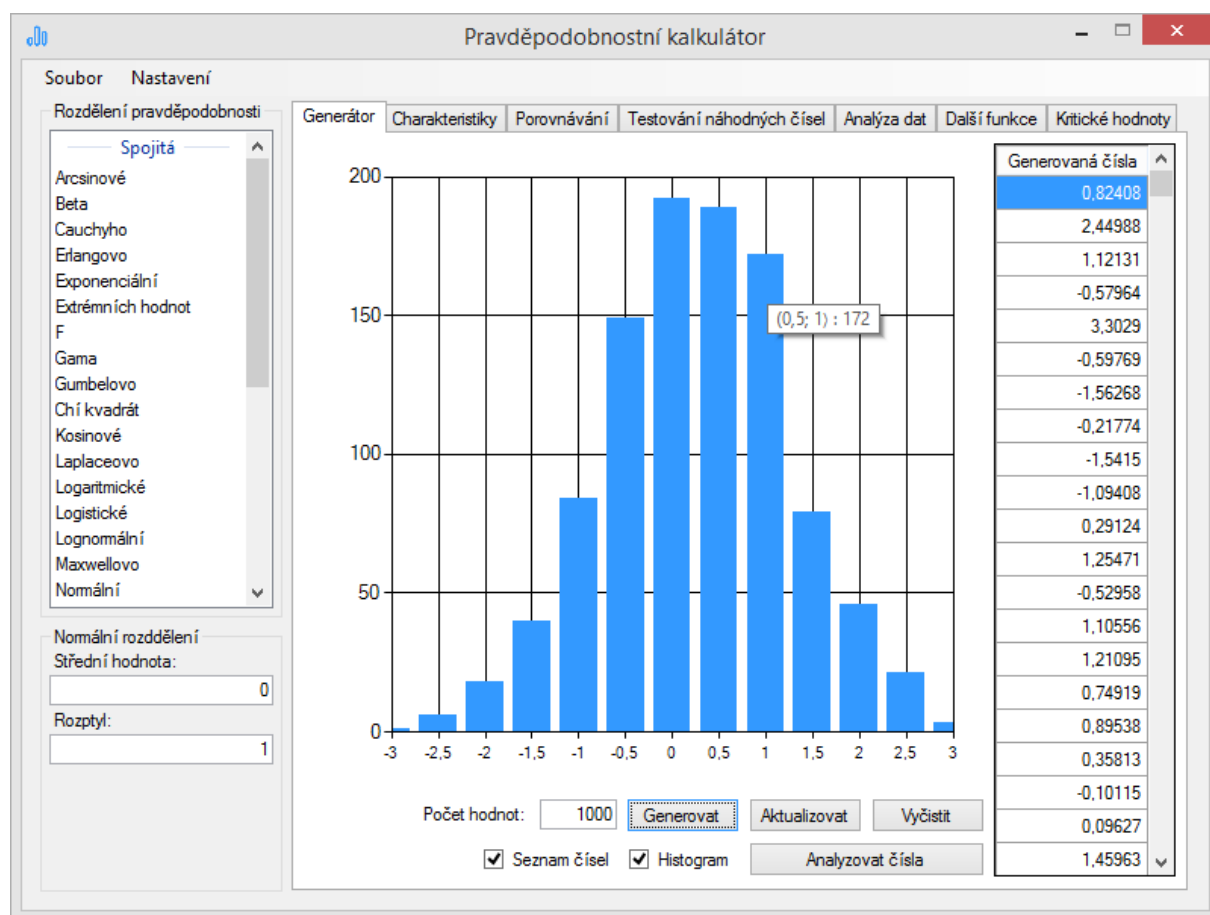


Obr. 22: Hlavní okno aplikace

V druhé části uživatel nalezne všechny implementované funkcionality. Každá karta pracuje nezávisle na ostatních. Některé k zajištění správného chodu spolupracují se seznamem rozdělení. Nastavení výstupů je pak zprostředkováno pomocí dialogových oken, která je možné spustit v již zmíněném menu nebo odpovídající klávesovou zkratkou.

2.1 Generátor

První zmíněnou funkcionalitou je generování hodnot náhodné proměnné. Nebudeme tedy zbytečně zdržovat a rovnou si ukážeme postup. Nejdříve vybereme požadované rozdělení pravděpodobnosti a určíme jeho parametry. Poté se rozhodneme, kolik čísel budeme potřebovat, zda se má vykreslit histogram či vypsát seznam hodnot a vyplníme příslušná pole. Po stisknutí tlačítka *Generovat* se spustí příslušný algoritmus a dojde ke generování čísel. V závislosti na zvolených možnostech se vykreslí histogram a vypíše seznam hodnot.

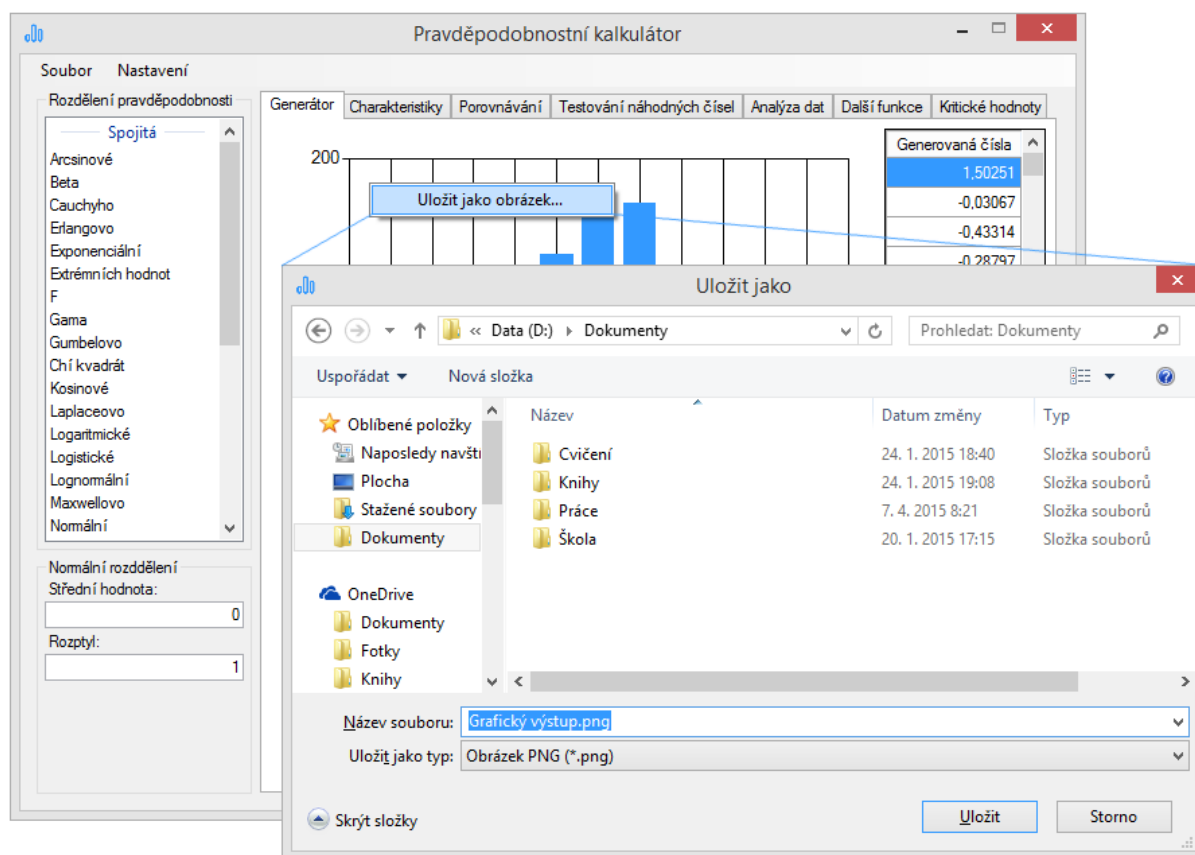


Obr. 23: Karta generování hodnot náhodné proměnné

Histogram je vytvořen buď automaticky, nebo dle uživatelem nastavených tříd. V případě automatického výpočtu kategorií je jejich počet získán ze vztahu $k = 1 + 3,3 \log n$ dle zdroje [5], kde k představuje počet tříd a n počet vygenerovaných čísel. Pokud chce uživatel třídy modifikovat, může je nastavit pomocí dialogového okna dostupného z hlavního menu

Nastavení > Histogram > Generování náhodné proměnné... nebo prostřednictvím kláves *Ctrl + H*. Samotnému nastavení je pak věnována zvláštní kapitola.

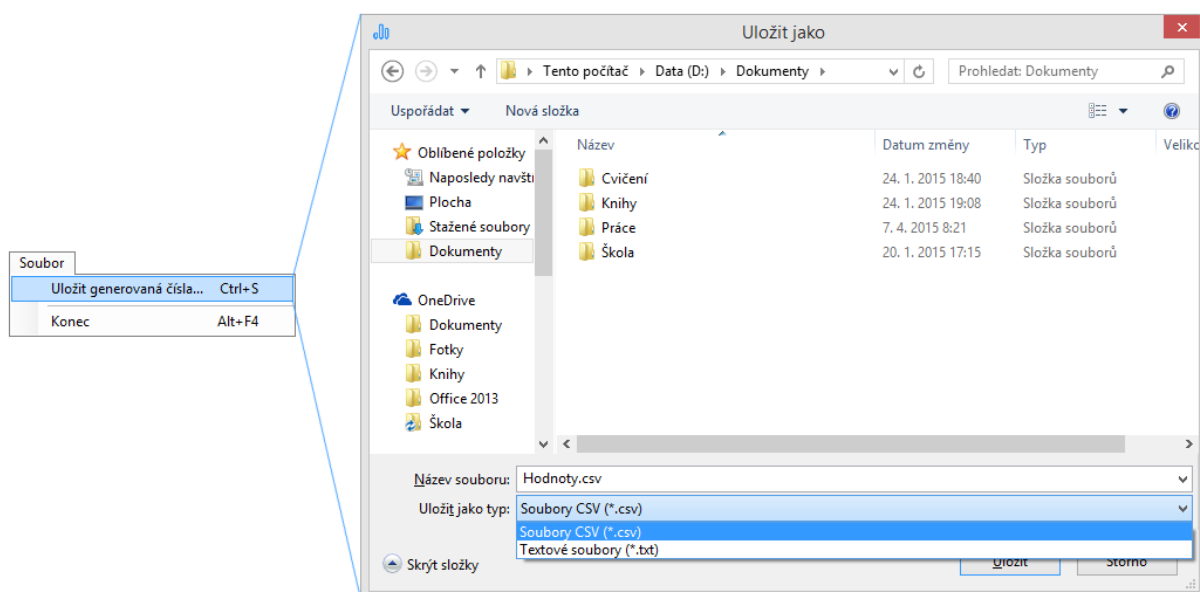
Při najetí myši na datový sloupec histogramu je zobrazena informace o hranicích třídy a četnosti prvků uvnitř, a to v následujícím tvaru: *(dolní mez, horní mez): četnost*. Histogram lze navíc uložit jako obrázek typu **.png*. Při stisknutí pravého tlačítka myši na ploše grafu se otevře kontextová nabídka, kde po stisknutí tlačítka dojde k vyžádání názvu a umístění souboru. Po potvrzení je obrázek uložen.



Obr. 24: Uložení grafického výstupu

Seznam hodnot obsahuje všechna vygenerovaná čísla a lze s ním pracovat obdobně jako s tabulkovými procesory. Pomocí *Ctrl + A* se označí všechny buňky, při vybírání myši se při stisknutí klávese *Ctrl* označují pouze jednotlivé prvky a při použití klávesy *Shift* pak celé rozsahy hodnot. Kombinací kláves *Ctrl + C* se označené hodnoty vloží do schránky a lze je pak použít mimo program. Počet desetinných míst lze nastavit pomocí níže zmíněného zaokrouhlování, *Nastavení > Zaokrouhlování...*, případně *Ctrl + Z*.

Vygenerované hodnoty se dají uložit také stisknutím klávesy *Ctrl + S* nebo pomocí menu následovně: *Soubor > Uložit generovaná čísla...* Uživatel je pak vyzván k zadání názvu a umístění výsledného souboru. Ten může být typu **.csv* nebo **.txt*. Jednotlivá čísla jsou v souboru umístěna v samostatných řádcích.



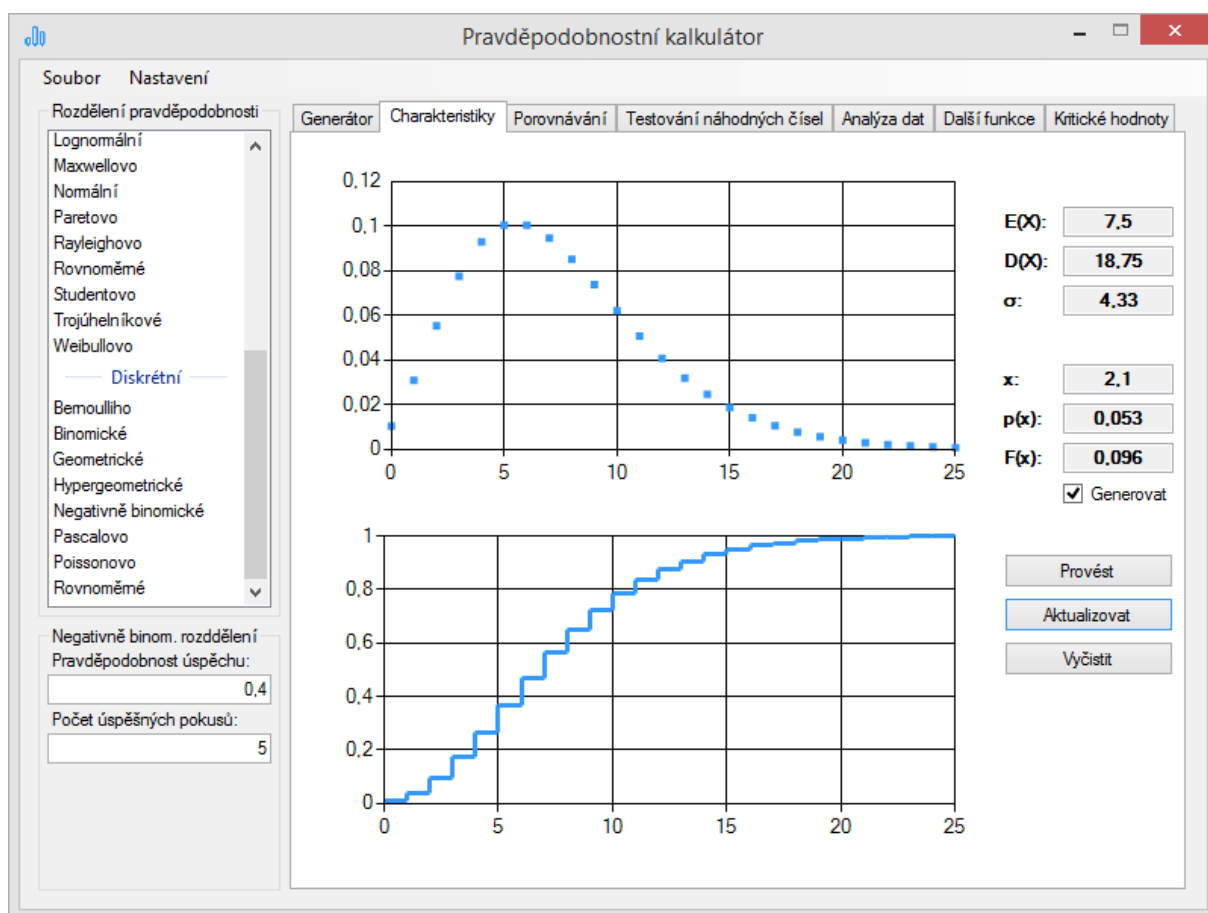
Obr. 25: Ukládání vygenerovaných hodnot

Nyní se podíváme na zbývající tlačítka. *Aktualizovat* slouží k opětovnému vykreslení histogramu a vypsání seznamu čísel dle nastavených hodnot. Nejsou tedy generována nová čísla, ale zobrazí se stávající hodnoty. Tento proces můžeme spustit i klávesou *F5*. Pomocí tlačítka *Vyčistit* se karta generátor vrátí do původního stavu. Smažou se vygenerované hodnoty, vyčistí se pole pro graf i seznam čísel. Poslední tlačítko *Analyzovat čísla* použije vygenerované hodnoty jako vstupní data pro analýzu, o které se budeme bavit v jedné z následujících kapitol.

Nakonec zmíníme možnost nastavení generátoru náhodných čísel. Jak víme z teoretické části, hodnoty náhodné proměnné jsou transformovány z rovnoměrně rozložených čísel na intervalu (0,1). A právě způsob jejich generování si také můžeme nastavit. Dialogové okno k tomu určené otevřeme buď pomocí kontextového menu *Nastavení > Generátor > Generování náhodné proměnné...* nebo kombinací kláves *Ctrl + G*. Možnosti nastavení jsou relativně rozsáhlé, a tak jim věnujeme zvláštní kapitolu.

2.2 Charakteristiky

Dále se zaměříme na kartu pro výpis a vykreslení charakteristik jednotlivých rozdělení pravděpodobnosti. Zde je postup následující: opět vybereme požadované rozdělení pravděpodobnosti a určíme jeho parametry, pak bez dalšího nastavování stiskneme tlačítko *Provést*. Během okamžiku vidíme jednotlivé výstupy. Nejdříve se zaměříme na ty textové.



Obr. 26: Karta charakteristik náhodné proměnné

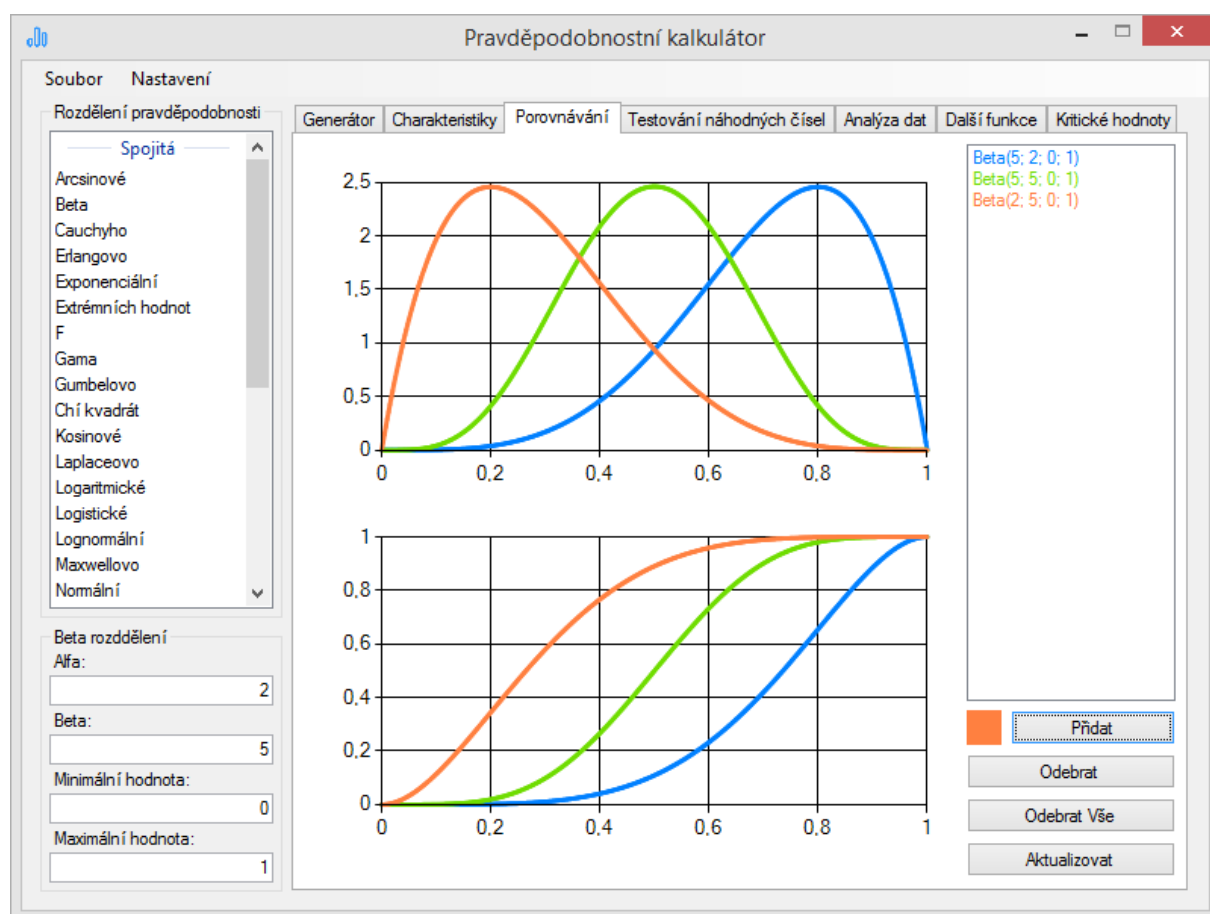
Použitá označení charakteristik korespondují s částí teoretický aparát a tvar výsledných hodnot je určen počtem desetinných míst. Ten můžeme nastavit po stisknutí tlačítka *Nastavení* > *Zaokrouhlování...* nebo klávesové zkratky *Ctrl* + *Z*. Nakonec musíme zmínit možnost generování hodnot hustoty pravděpodobnosti či pravděpodobnostní funkce a distribuční funkce na základě pozice kurzoru myši na ploše grafu. Dochází k tomu pouze tehdy, je-li zaškrtnuto tlačítko *Generovat*. V opačném případě je možné zadat hodnotu x ručně. Klávesou *Enter* či vyskočením z pole pro zadání je x potvrzeno a dojde k vlastnímu výpočtu.

Grafickým výstupem ve formě grafu je pak hustota pravděpodobnosti či pravděpodobnostní funkce v horní a distribuční funkce ve spodní části. Jedná-li se o spojitá rozdělení, je použit spojnicový graf, u diskretních pak bodový. V případě potřeby jej můžeme uložit jako obrázek, postup je stejný jako u histogramu na kartě *Generátor*. Hodnoty na osách či tloušťku čáry je možné nastavit v menu *Nastavení* > *Graf* > *Charakteristiky...* nebo po použití klávesové zkratky *Ctrl* + *K*. Barvu datové řady můžeme změnit v části *Nastavení* > *Barvy...* nebo po stisknutí kombinace kláves *Ctrl* + *B*. Možnosti nastavení jsou podrobně popsány v odpovídající kapitole.

Tlačítko *Aktualizovat* má obdobný význam jako na kartě *Generátor* a také místo něj můžeme použít klávesu *F5*. Stisknutím se překreslí graf a přepíše charakteristiky dle požadovaného nastavení. *Vyčistit* pak uvede kartu do původního stavu.

2.3 Porovnávání

Další velice zajímavou a užitečnou funkcí je možnost porovnávání různých rozdělení pravděpodobnosti. I zde začneme zvolením libovolného rozdělení pravděpodobnosti a nastavením jeho parametrů. Poté vybereme barvu, kterou bude vykresleno, a klikneme na tlačítko *Přidat*. Do grafu je ihned vložena nová datová řada a v seznamu se objeví název rozdělení a hodnotami jeho parametrů. Pro přidání dalšího rozdělení stačí postup zopakovat.



Obr. 27: Karta porovnávání náhodné proměnné

Výstup na této kartě pak představuje graf složený z několika datových řad. I ten můžeme uložit jako obrázek, postup je stále stejný jako u předešlých grafických výstupů. Samozřejmě je možné libovolně nastavovat osy a tloušťku čar. Slouží k tomu dialogové okno, jež můžeme otevřít v menu *Nastavení > Graf > Porovnávání...* či klávesovou zkratkou *Ctrl + P*. Dialog pro nastavení spojnicových, případně bodových grafů je rozebrán níže.

V seznamu můžeme klikáním jednotlivá rozdělení označit, a to buď po jednom, nebo více zároveň. S klávesou *Ctrl* jsou rozdělení označována postupně, s tlačítkem *Shift* pak

vybíráme celé rozsahy. Takto zvolená rozdělení pak můžeme velice snadno odstranit, a to buď pomocí tlačítka *Odebrat* nebo stisknutím klávesy *Delete*.

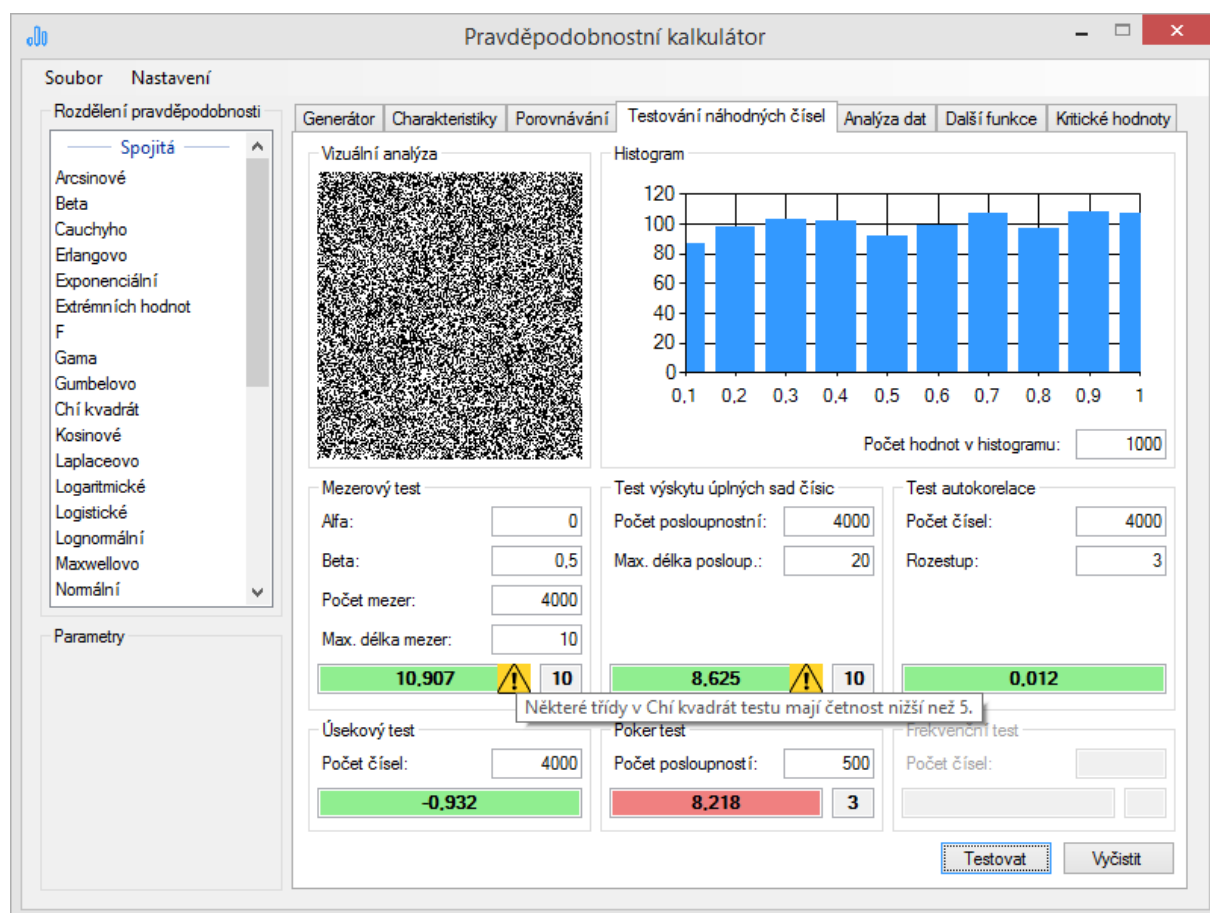
Odebrat vše pak smaže všechna rozdělení ze seznamu a vyčistí graf. Tlačítko *Aktualizovat* a klávesa *F5* mají podobný význam jako na předešlých kartách, dojde k překreslení grafu dle potřeb uživatele.

2.4 Testování náhodných čísel

Nově lze, oproti aplikaci z bakalářské práce, testovat náhodná čísla. Zde začneme zvolením generátoru náhodných čísel. K tomu slouží speciální dialogové okno dostupné přes menu *Nastavení > Generátor > Testování náhodných čísel...* nebo kombinaci kláves *Ctrl + E*. Implicitně je nastaven generátor platformy .NET. Dále můžeme použít lineární a kvadratický kongruenční generátor s libovolně zvolenými parametry a samozřejmě i seznam vlastních hodnot. Samotné nastavení je rozebráno v kapitole 2.8.1.

Nyní si zvolíme, které testy chceme provést a nastavíme si jednotlivé hladiny významnosti. Slouží k tomu samostatné okno dostupné z menu *Nastavení > Testy náhodnosti...* nebo pomocí klávesové zkratky *Ctrl + T*. I tento dialog je popsán samostatně, a to v kapitole 2.8.4.

Ted' už zbývá jen nastavit parametry jednotlivých testů. Abychom dostávali výsledky s dobrou vypovídající hodnotou, musíme postupovat s rozvahou. Parametry je potřeba určit tak, aby testy měly dostatečné množství dat. To může být například u χ^2 -testu problém. Ale platí zde i jiná pravidla. Je tedy nutné disponovat alespoň základními teoretickými znalostmi. K tomu je určena kapitola v části teoretický aparát a doporučená literatura uvedena tamtéž. Máme-li vyplněna všechna vstupní pole, můžeme testování spustit pomocí tlačítka *Testovat*. Během chvíle se karta zaplní a my můžeme pokračovat ve vyhodnocování generátoru či sekvence čísel. Jednotlivé výstupy si popíšeme nyní. Ukázka zobrazení vypnutého testu je demonstrována na frekvenčním testu.



Obr. 28: Karta testování náhodné proměnné

První dva výstupy jsou grafické a slouží k prvotnímu zhodnocení. Princip vizuální analýzy je zmíněn v teoretické části, a tak zde uvedeme jen možnost jejího uložení do souboru typu *.png. To umožňuje i histogram. Postup je pak stejný jako u ostatních grafických výstupů zmíněných výše. Třídy histogramu na této kartě nelze měnit a jsou pro všechna testování stejná. Při najetí myši na datový sloupec je zobrazen popisek stejně jako u histogramu na kartě *Generátor*.

Další testy již dávají přímo výslednou hodnotu, tedy hodnotu testovacího kritéria. V případech, kdy se jedná o χ^2 -test, je navíc uvedeno textové pole s počtem stupňů volnosti. Podrobný popis jednotlivých testů je uveden v teoretické části této publikace. Aby nebylo nutné hledat v odpovídajících tabulkách kritických hodnot, aplikace sama výsledek testu vyhodnotí. V případě platnosti nulové hypotézy je výsledná kritická hodnota podbarvena zeleně, při nesplnění červeně. Pokud test nedává důvěryhodné výsledky, je na to uživatel upozorněn výstražným trojúhelníkem. Ten při najetí myši zobrazí důvod snížené důvěry.

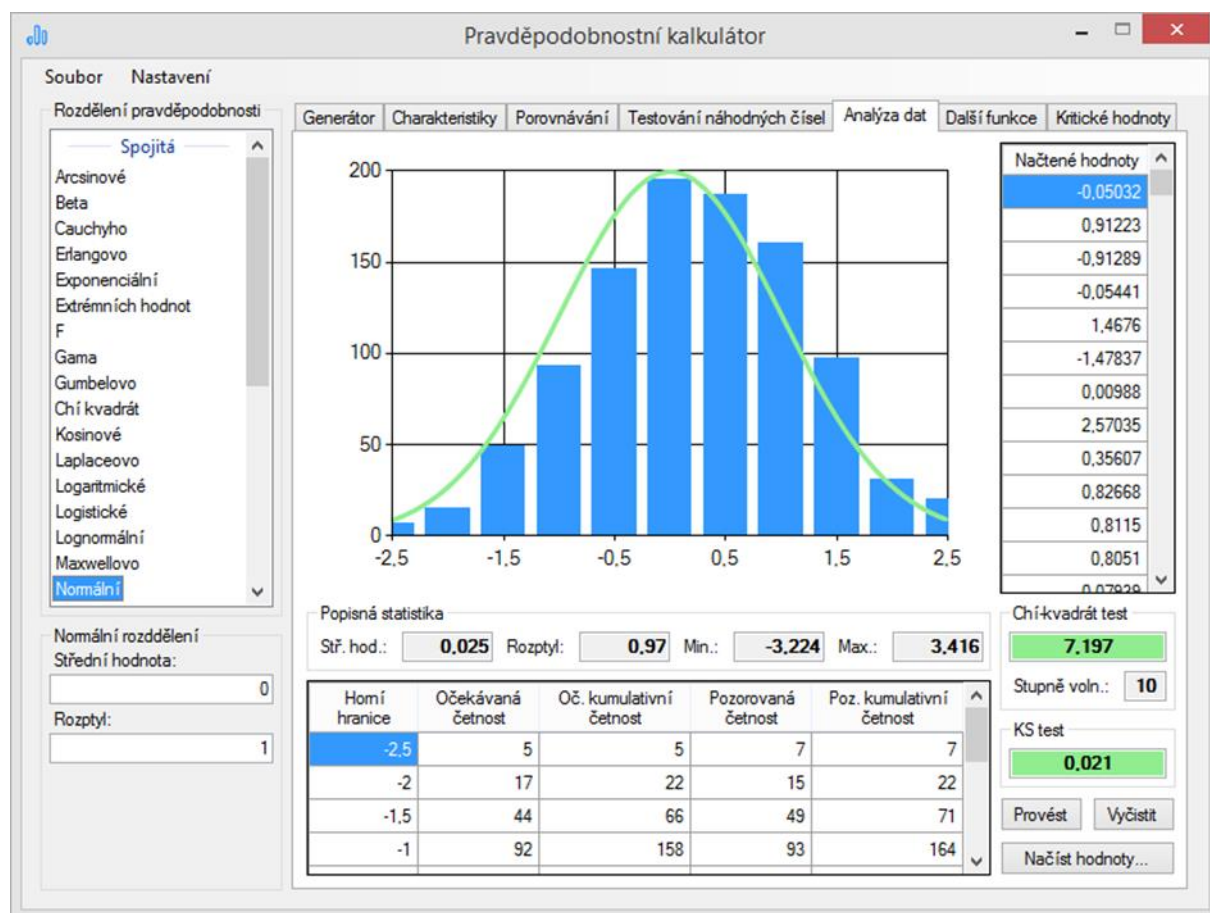
Pokud by uživatel přece jen chtěl znát hodnotu testovacího kritéria, může pro obecné testy použít kartu *Kritické hodnoty*, která je rozebrána v kapitole 2.7. Výše zmíněné podbarvení lze samozřejmě změnit, slouží k tomu dialogové okno *Barvy*, které je dostupné z menu

Nastavení > *Barvy...* nebo pomocí klávesové zkratky *Ctrl + B*. To se hodí například v situaci, kdy nemáme na zobrazovacím zařízení kalibrované barvy, špatně zobrazené podbarvení by pak mohlo být matoucí.

Už zbývá jen zmínit tlačítko *Vyčistit*, jež vrátí kartu pro testování do původního stavu, vymaže tedy všechny zobrazené výsledky.

2.5 Analýza dat

Další novinkou je možnost analýzy naměřených či jinak získaných dat. Zde máme několik možností, jak postupovat. Buď nejdříve načteme hodnoty a na základě nich vybereme rozdělení pravděpodobnosti, které chceme proložit, nebo rovnou vybereme rozdělení a po dotázání vložíme naměřená data.



Obr. 29: Karta analýzy dat

U první možnosti lze data získat dvěma způsoby. Buď je vygenerovat na kartě *Generátor* a poté přenést pomocí tlačítka *Analyzovat čísla*, nebo je načíst ze souboru. K tomu slouží tlačítko *Načíst hodnoty...* karty *Analýza dat*. Soubor hodnot musí být ve formátu *.csv nebo *.txt, oddělovačem je desetinná čárka a jednotlivé hodnoty jsou na řádcích uvedeny samostatně. Jedná se tedy o stejné požadavky jako na soubor náhodných čísel, který je uveden

níže, pouze s tím rozdílem, že hodnoty mohou být libovolné. Nejsou tedy omezeny žádným intervalem.

Po úspěšném načtení dat se vykreslí histogram, vypíše seznam čísel a popisná statistika. Uživatel tedy může před zvolením rozdělení pravděpodobnosti naměřené hodnoty nejdříve analyzovat. Pokud má vše rozmyšleno, vybere rozdělení pravděpodobnosti, nastaví jeho parametry a stiskne tlačítko *Provést*. Pokud hodnoty doposud nenačetl, bude k tomu vyzván nyní, což je druhá z možností.

Proběhlo-li vše v pořádku, je histogram proložen požadovaným rozdělením pravděpodobnosti, je vypsána tabulka s výsledky a jsou provedeny testy shody. Nejdříve zmíníme tabulku s četnostmi. Ta obsahuje sloupce s horními hranicemi tříd a očekávané i pozorované četnosti. U očekávaných četností mohou být výsledky nepatrně zkresleny zaokrouhlováním. To se projeví především u kumulativní očekávané četnosti.

Dále se budeme věnovat testům shody. Výpis výsledků probíhá obdobně jako na kartě *Testování náhodných čísel*. U χ^2 -testu může nastat situace, že je v nějaké třídě očekávaná četnost rovna nule, v tomto případě je vypsáno chybové hlášení a test neproběhne. Došlo by totiž k dělení nulou. Stane-li se, že v některé z tříd je četnost nižší než 5, zobrazí se varování. V těchto případech je vhodné nastavit histogram ručně tak, aby byla splněna doporučení pro daný test. Jinak by mohlo docházet ke zbytečnému zamítnutí platné nulové hypotézy. Nastavení histogramu je možné provést v okně dostupném z menu *Nastavení > Histogram > Analýza dat...* či klávesovou zkratkou *Ctrl + L*.

Nakonec zmíníme samotné prokládání grafu. Pokud u obou testů dojde k potvrzení platnosti nulové hypotézy, je vykreslena zeleně. V opačném případě červeně. Barvy je možné změnit stejně jako u výsledků na kartě *Testování náhodných čísel*.

2.6 Další funkce

Na této kartě můžeme získat výsledky hned 8 zajímavých funkcí. Každou si probereme samostatně hned potom, co zmíníme společné vlastnosti a možnosti polí s výsledky. V případě, že je výsledkem desetinné číslo, můžeme nastavit jeho zaokrouhlení. To se provede pomocí speciálního dialogového okna popsaného níže, které můžeme otevřít z menu *Nastavení > Zaokrouhlování...* nebo stisknutím kláves *Ctrl + Z*. Pokud je výsledkem příliš velké číslo, které se nevejde do výstupního pole, můžeme se v něm pohybovat pomocí myši či kurzorových kláves. Také jej můžeme označit a vykopírovat klasickým způsobem, pomocí *Ctrl + C*. Pokud se místo výsledku vypíše *nekonečno*, znamená to, že je číslo příliš velké a „nevešlo se“ do

použitého datového typu. Další poznámky podobného typu případně zmíníme u odpovídajících funkcí.

Obr. 30: Karta s dalšími funkcemi

Charakteristiky a funkce náhodné veličiny vyžadují jako jedině na této kartě zvolení požadovaného rozdělení pravděpodobnosti a určení jeho parametrů. V případě výpočtu charakteristických funkcí dále musíme zadat hodnotu x . Jedná se o „štíhlejší“ variantu karty *Charakteristiky*. Pokud nás zajímají pouze konkrétní hodnoty, pak tato možnost výpočtu bohatě stačí.

Následuje faktoriál, u něj je potřeba zmínit snad jen možnosti jeho výpočtu. Při zaškrtnutém políčku *Aproximace* je výsledek získán následovně. Pokud je $n \leq 170$, probíhá výpočet klasickým postupem, tedy dle vzorce uvedeného v teoretické části. Je-li ovšem $n > 170$, výslednou hodnotu získáme pomocí přirozeného logaritmu. Výsledná hodnota sice není přesná, ale můžeme faktoriál využít pro mnohem větší n . Samotný výpočet uživatel v případě zájmu nalezne ve zdroji [13]. Výsledkem tak může být i číslo v následujícím tvaru: $1,0E + 10$. Zůstane-li políčko *Aproximace* nezaškrtnuto, použije se pro výpočet datový typ `BigInteger`, který je omezen pouze velikostí operační paměti počítače. Výpočet se provede dle explicitního vzorce, může sice trvat delší dobu, ale výsledek je pak vždy přesný.

U chybové funkce není potřeba nic rozepisovat, zadáme hodnotu x , klikneme na tlačítko *Provést* a ihned se vypíše výsledek. Ani kombinační číslo a Stirlingovo číslo 2. druhu není třeba podrobně popisovat, pouze zadáme hodnoty a necháme aplikaci pracovat. Okamžitě se vypíše výsledek. Smysl tlačítka *Aproximace* je obdobný jako u faktoriálu, není jej tedy potřeba znovu zmiňovat. Bližší informace je možné nalézt ve zdroji [13].

Konečně se dostáváme k beta a gama funkci. I zde stačí pouze zadat požadované hodnoty parametrů funkcí. Pokud chceme vypočítat neúplnou beta či gama funkci, stačí zaškrtnout tlačítko *Neúplná* a vyplnit hodnotu x . Stisknutím tlačítka *Provést* proběhne výpočet a vypíše se výsledek.

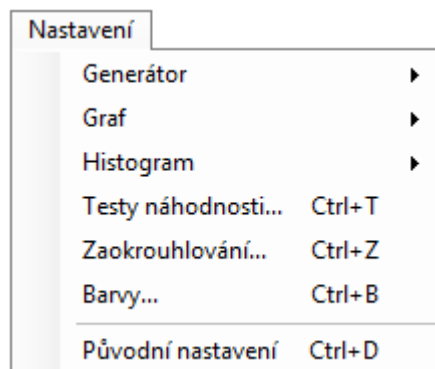
2.7 Kritické hodnoty

Poslední karta je určena pro získání kritických hodnot nutných k vyhodnocení některých statistických testů. Nemusíme tedy hledat žádné tabulky, učebnice či jiné publikace. Získáme je velice snadno pomocí tohoto nástroje. Zadáme pouze požadované parametry a stiskneme tlačítko provést. Výsledná kritická hodnota je okamžitě zobrazena.

Obr. 31: Karta s kritickými hodnotami

2.8 Nastavení

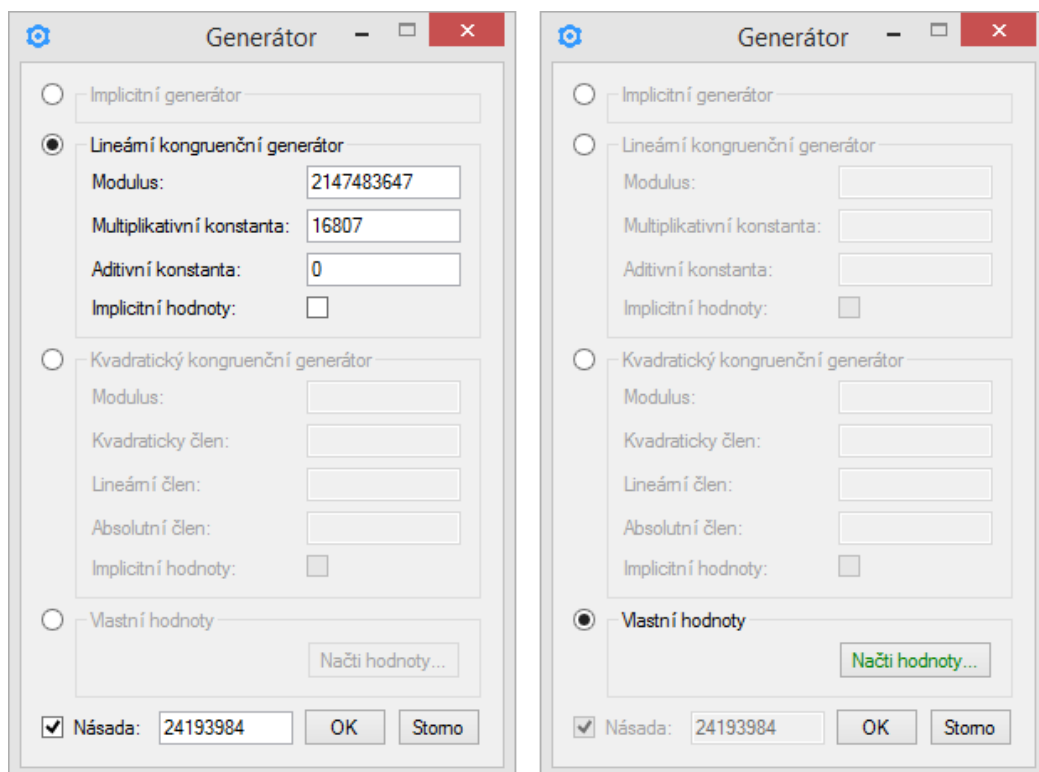
V této části se zaměříme na položky menu *Nastavení*. Pečlivě si probereme dialogová okna odpovídající jednotlivým možnostem nastavení a vysvětlíme si, jak s nimi pracovat. Všechna tato okna je možné ukončit i pomocí kláves *Enter* a *Escape*. *Enter* slouží k potvrzení provedených úprav, *Escape* pak k jejich stornování. Při otevření některého z oken jsou vždy zobrazeny hodnoty aktuálního nastavení. Začneme vrácením nastavení do původního stavu. K tomu slouží tlačítko *Původní nastavení* či kombinace kláves *Ctrl + D*.



Obr. 32: Kontextová nabídka menu pro nastavení

2.8.1 Generátor

První položka slouží k nastavení generátoru, a to buď pro generování hodnot náhodné proměnné či analýzu dat. Samotné nastavení se provádí pomocí následujícího formuláře. My si teď jednotlivé možnosti podrobněji rozebereme.

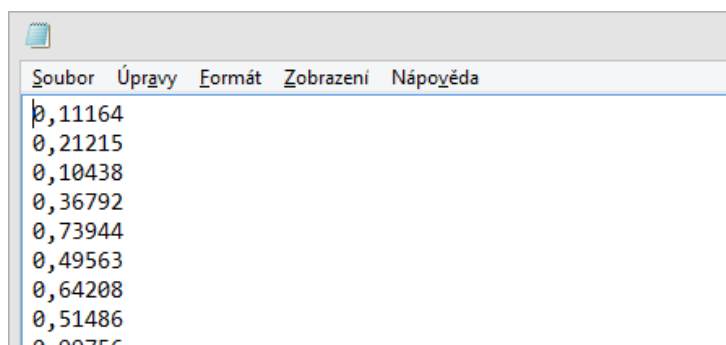


Obr. 33: Dialogové okno pro nastavení generátoru

První a nejjednodušší možností je použití generátoru pseudonáhodných čísel platformy .NET. U něj můžeme buď použít vlastní násadu, nebo ji nechat nastavit automaticky. Pokud ji zvolíme ručně, můžeme opakovaně generovat stejnou posloupnost čísel.

Dále je možné využít lineární a kvadratický kongruenční generátor. U nich si můžeme nastavit parametry buď sami, nebo použít implicitní. Vstupní hodnoty musí samozřejmě splňovat podmínky uvedené v teoretické části a vždy musí být vyplněny všechny.

Poslední možností je využití vlastní kolekce čísel. Tu je možné načíst z textových souborů typu *.csv a *.txt. Musí být ve formátu, kdy je každá hodnota uvedena samostatně na řádku a jako oddělovač je použita desetinná čárka – viz obrázek. Samozřejmostí je, že hodnoty musí náležet intervalu $\langle 0; 1 \rangle$. Při úspěšném načtení hodnot se barva textu tlačítka změní na zelenou, pokud nastane chyba, text se zobrazí červeně.



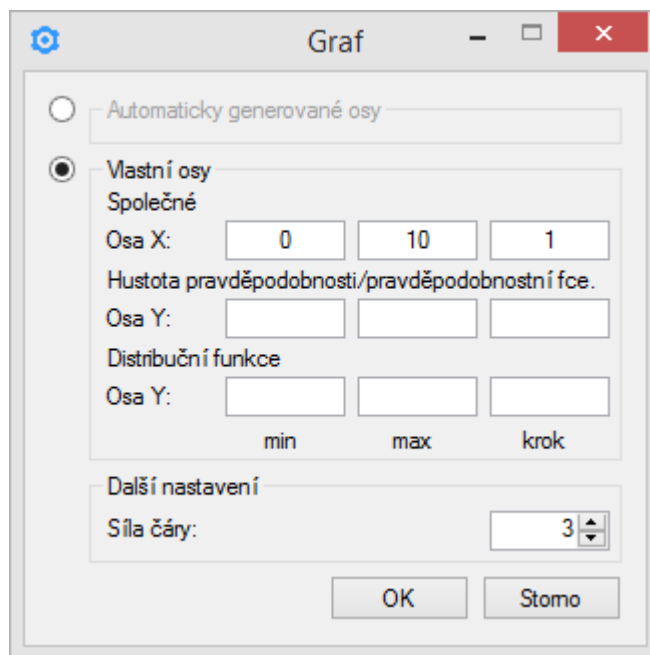
Obr. 34: Formát souboru náhodných čísel

Musíme ještě upozornit na fakt, že generátor je vytvořen pouze při potvrzení hodnot tohoto dialogu. Pokud tedy chceme reprodukovat posloupnost čísel, musíme před samotným generováním znovu otevřít dialogové okno pro nastavení generátoru a potvrdit stávající hodnoty. Tím se vytvoří nová instance generátoru. Pokud by se tak nestalo, pokračovali bychom v generování tam, kde jsme naposledy skončili.

2.8.2 Graf

Tato část slouží k nastavení os grafu a síle vykreslených datových řad. Okno je stejné jak pro zobrazení jednotlivých charakteristik, tak i pro jejich porovnávání. Začneme tloušťkou čáry, ta může být nastavena od 1 do 5.

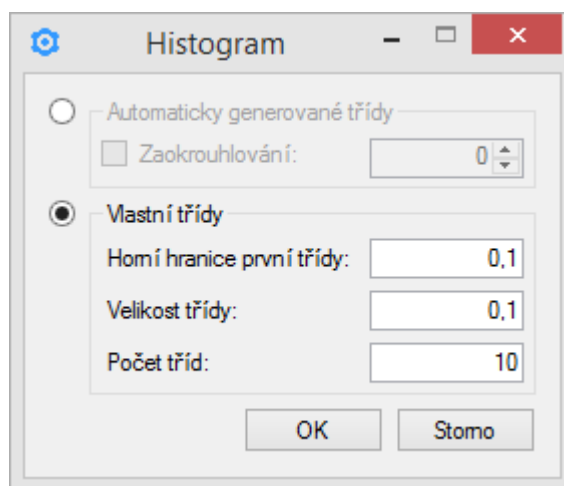
Osy můžeme nechat generovat automaticky. Potřebujeme-li však zobrazit pouze určitý rozsah hodnot nebo nějaký detail, máme možnost si je nastavit ručně. Osa x je pak společná pro oba vykreslené grafy a osa y můžeme určit samostatně. Vždy však musí být vyplněn celý řádek, tedy minimální i maximální hodnota a krok. Pokud řádek vyplněn není, je příslušná osa nastavena automaticky.



Obr. 35: Dialogové okno pro nastavení grafu

2.8.3 Histogram

U nastavení histogramu je potřeba zmínit, že se nejedná o způsob jeho vykreslení, ale o určení mezí jednotlivých tříd. To můžeme udělat několika způsoby. První možností je nechat třídy generovat automaticky a pouze nastavit, na kolik desetinných míst mají být zaokrouhleny. Pokud zaokrouhlovat nechceme, ponecháme tlačítko *Zaokrouhlování* nezaškrtnuté. Druhým způsobem je ruční nastavení, kde si určíme horní hranici první třídy, velikost jedné třídy a nakonec celkový počet těchto tříd. Dialogové okno vypadá stejně jak pro generování hodnot, tak analýzu dat.

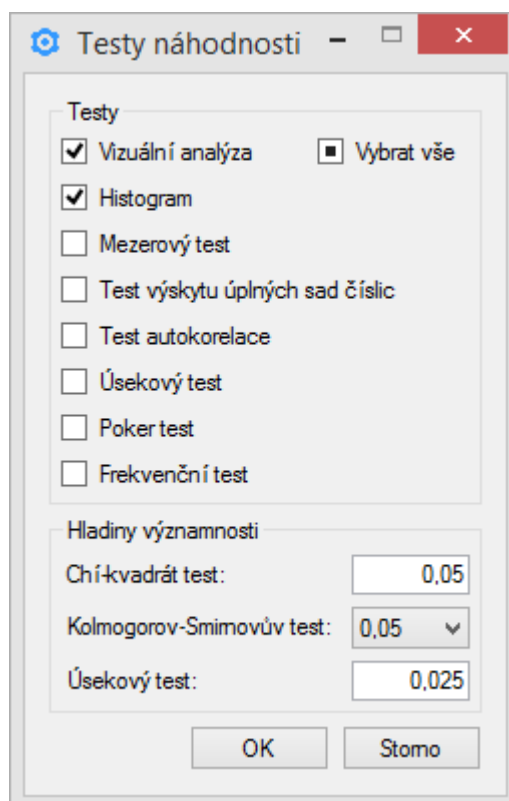


Obr. 36: Dialogové okno pro nastavení tříd histogramu

2.8.4 Testy náhodnosti

Na tomto místě můžeme určit, které testy náhodnosti se mají provést a které nikoliv. Dále je zde možné nastavit hladiny významnosti jednotlivých obecných testů. Protože se jedná

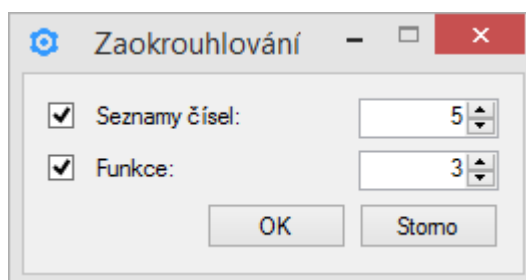
o hladiny významnosti α , připomeneme, že požadovaný kvantil je pak určen pomocí vztahu $1 - \alpha$. Tlačítko *Vybrat vše* funguje klasickým způsobem.



Obr. 37: Dialogové okno pro nastavení testů náhodnosti

2.8.5 Zaokrouhlování

Zaokrouhlování můžeme nastavit pro seznamy čísel a výstupy funkcí. Seznamy čísel se vyskytují v části pro generování hodnot náhodné proměnné a analýzu dat. Výstupy funkcí pak můžeme nalézt na většině karet. Většinou se jedná o šedá textová pole, která nejsou zpřístupněna pro zápis.

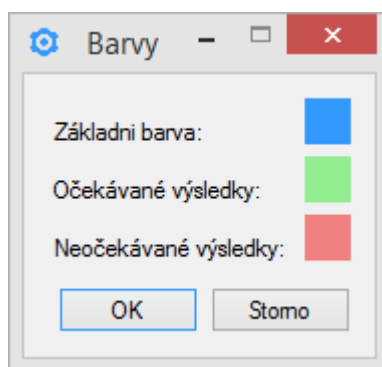


Obr. 38: Dialogové okno pro nastavení zaokrouhlování

Hodnoty uvedené na obrázku představují implicitní nastavení a udávají výsledný počet desetinných míst. Jedná se o zaokrouhlení vypsanych čísel, nikoliv hodnot jako takových. Do dialogového okna se pak dostaneme pomocí tlačítka *Zaokrouhlování...* v menu nebo klávesovou zkratkou *Ctrl + Z*.

2.8.6 Barvy

Jako poslední zmíníme dialogové okno určené pro nastavení barev. První položka udává barvu datových řad spojnicových, případně bodových grafů na kartě charakteristiky a také datových sloupců všech histogramů. Další dvě jsou určeny k rozlišení výsledků statistických testů. Pokud platí nulová hypotéza, je k zvýraznění použita barva pro očekávané výsledky, v opačném případě barva pro výsledky neočekávané.



Obr. 39: Dialogové okno pro nastavení základních barev

Dialogové okno pro nastavení barev je dostupné z menu stisknutím tlačítka *Barvy...*, případně kombinací kláves *Ctrl + B*.

3 STRUKTURA APLIKACE

Obsahem této kapitoly je přiblížení vnitřní struktury programu. Jedná se o aplikaci napsanou v čistě objektovém programovacím jazyce C# za použití frameworku .NET 4.5.1. Struktura celého projektu je inspirována architekturou MVC (Model-View-Controller), kterou si nyní stručně popíšeme.

Smyslem této architektury je rozdělení systému do tří nezávislých vrstev. Jsou tak striktně oddělena klientská rozhraní od samotné logiky aplikace i přístupu k datům. To je vhodné především pro rozsáhle projekty, při nichž je umožněno bez větších problémů vyměnit jednu část za jinou, aniž bychom museli provádět rozsáhlé úpravy na ostatních vrstvách. Jako jednoduchý příklad může sloužit změna formuláře aplikace či aktualizace formátu výstupního datového souboru.

Vrstva **Model** obsahuje vlastní logiku aplikace. Její funkce tedy spočívá v přijetí požadavku z okolí a případnému vrácení požadovaných informací. Část **View** je určena pro komunikaci s uživatelem. Slouží nejen k zadávání požadavků, ale i k prezentaci požadovaných výstupů. A nakonec **Controller**, který je zodpovědný za spojení všech částí dohromady. Mimo to reaguje na požadavky a připravuje výsledné pohledy (view).

Tímto přístupem jsem se tedy inspiroval a použil jednu z jeho modifikací. Celý program jsem rozdělil na vrstvy Control, Entity a Presentation, jejichž význam je následující. V části **Control** je umístěna tzv. logika aplikace, tedy různé matematické výpočty. Vrstva **Entity** pak představuje klíčové objekty systému, které jsou dále zpracovávány. Jedná se například o třídu zapouzdřující histogram nebo různé výčtové typy. Interakce s uživatelem je pak zajištěna pomocí **Presentation**. Zde jsou umístěny formuláře, pomocí kterých jsou zadávány požadavky a zobrazovány požadované výstupy. Nyní si jednotlivé části stručně představíme.

3.1 Control

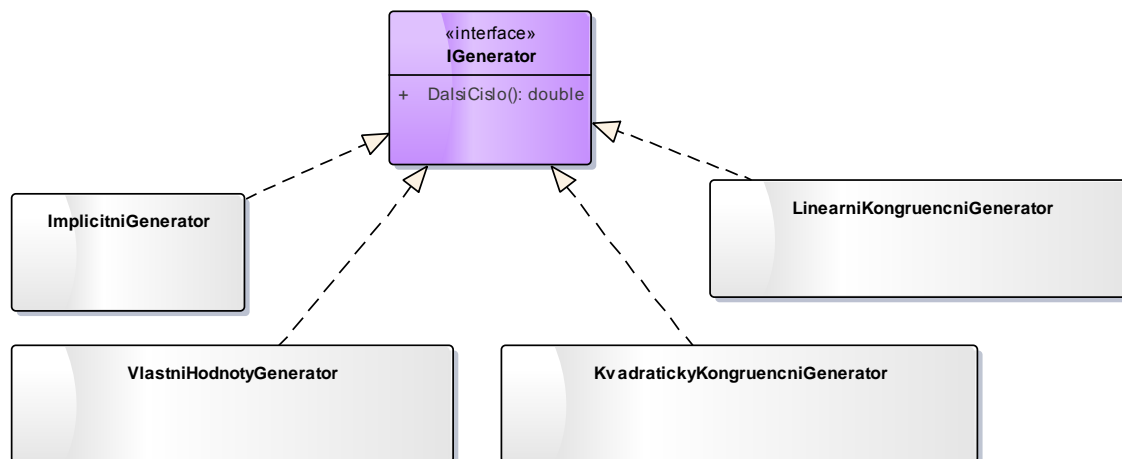
Jak již bylo zmíněno, tato část slouží pro nejrůznější výpočty. Jsou zde umístěny algoritmy pro generování pseudonáhodných čísel, jejich testování a transformace na různá rozdělení pravděpodobnosti, výpočty charakteristik těchto rozdělení a důležité matematické funkce. Díky použití rozhraní je aplikace velice snadno škálovatelná.

3.1.1 Generátory pseudonáhodných čísel

Všechny generátory pseudonáhodných čísel implementují rozhraní `IGenerator`, které obsahuje signaturu pouze jedné metody, získání dalšího čísla. Parametry prvních třech generátorů je možné nastavit pomocí konstruktoru. Můžeme využít implicitní, který je bezparametrický, případně implicitní s násadou, kde zadáváme pouze hodnotu tzv. semínka.

Kongruenční generátory jsou doplněny o konstruktor umožňující měnit jejich vnitřní nastavení, tedy konstanty použité v rekurentním předpisu pro získávání hodnot.

Poslední „generátor“ je určen pro použití uživatelem vložených hodnot. Má pouze jeden konstruktor, jehož parametrem je kolekce hodnot implementující generické rozhraní `IEnumerable<double>`. Struktura je naznačena na zjednodušeném diagramu tříd.

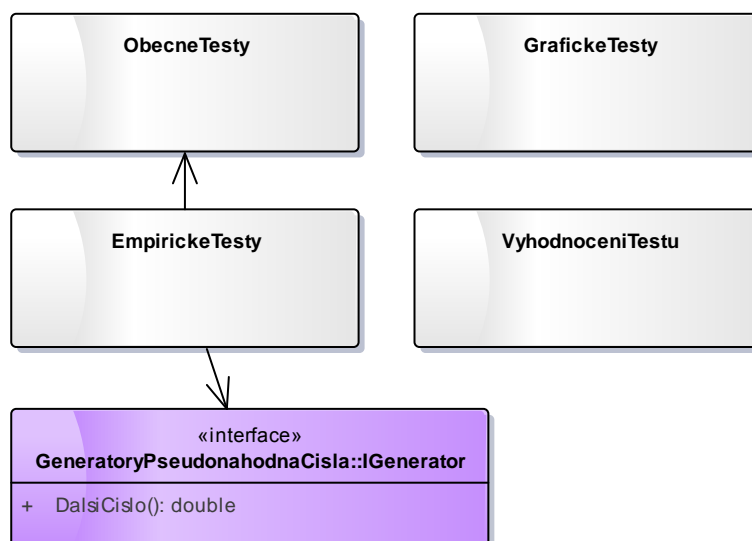


Obr. 40: Zjednodušený diagram tříd – generátory pseudonáhodných čísel

Rekurentní vzorce pro kongruenční generátory jsou inspirovány zdroji [3], [4] a [5].

3.1.2 Testy náhodných čísel

Použité testy jsou rozděleny do třech skupin – obecné, empirické a grafické. Tato část je navíc doplněna o třídu sloužící k vyhodnocení jednotlivých testů. Strukturu zachycuje zjednodušený diagram tříd.



Obr. 41: Zjednodušený diagram tříd – testy náhodných čísel

Třída s obecnými testy obsahuje metody pro výpočet χ^2 a Kolmogorovova-Smirnovova testu. Ty jsou pak používány buď samostatně, nebo prostřednictvím empirických testů.

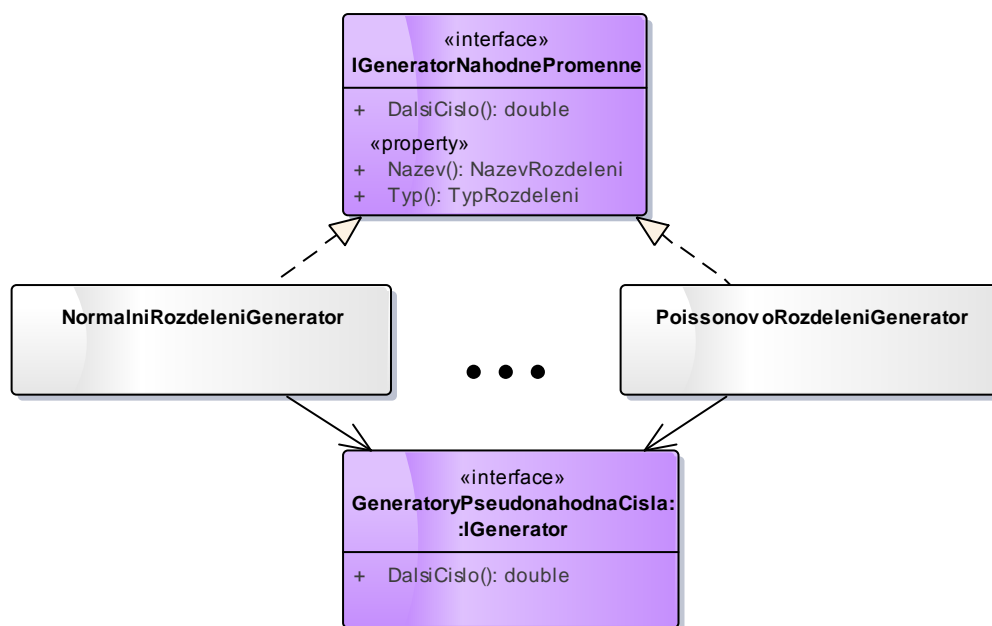
Výsledkem je vždy kritická hodnota datového typu `double`. Pro vyhodnocení kritických hodnot je použita statická třída, která na základě použitého testu a jeho parametrů potvrdí či vyvrátí nulovou hypotézu. Výsledkem je hodnota typu `bool`. Empirické testy mají konstruktor s parametrem typu `IGenerator`, který představuje zkoumaný generátor. Obecné testy mají konstruktor bezparametrický a vstupní hodnoty jsou zadávány v podobě parametrů metod pro výpočet kritických hodnot. Třída zapouzdřující grafické testy je statická, a tak konstruktor nemá. Obsahuje pouze metodu `VizualniAnalyza` s návratovou hodnotou typu `Image`.

Algoritmy pro obecné a empirické testy byly vytvořeny na základě matematických předpisů ze zdrojů [2], [3], [7] a [8].

3.1.3 Transformace náhodných čísel

Pro generování hodnot náhodné proměnné slouží třídy, které transformují čísla získaná pomocí generátorů pseudonáhodných čísel. Třídy zajišťující transformaci implementují rozhraní `IGeneratorNahodnePromenne`, jež obsahuje signatury vlastností určující typ a název rozdělení (viz výčtové typy níže) a metody pro získání dalšího čísla.

Kromě metod rozhraní pak konkrétní třídy obsahují konstruktor s parametry určujícími vlastnosti požadovaného rozdělení a samotným generátorem pseudonáhodných čísel typu `IGenerator`. Algoritmy výpočtu jsem získal buď metodou inverzní transformace, nebo použitím předpisů ze zdrojů [2] a [14]. Ty jsem v několika případech doplnil například o parametr posunutí či možnost „roztážení“ na požadovaný interval. Struktura je naznačena na zjednodušeném diagramu tříd.

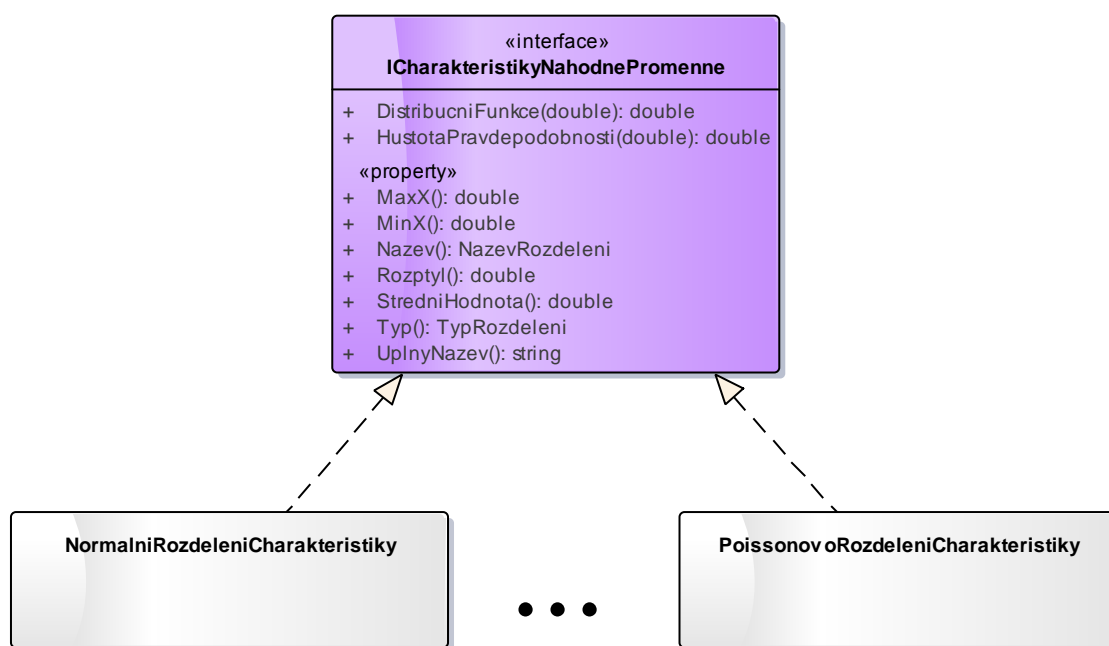


Obr. 42: Zjednodušený diagram tříd – transformace náhodných čísel

3.1.4 Charakteristiky

Aby bylo možné vypsát a vykreslit charakteristiky jednotlivých rozdělení, bylo nutné vytvořit třídy zapouzdřující požadované výpočty. Všechny tyto třídy implementují rozhraní `ICharakteristikyNahodnePromenne`, které disponuje následujícími signaturami. Metody pro výpočet hustoty pravděpodobnosti (pravděpodobnostní funkce) a distribuční funkce s parametrem typu `double` představujícím hodnotu x . Dále vlastnosti vracející střední hodnotu a rozptyl. Pro potřeby aplikace byly přidány další vlastnosti jako název a typ rozdělení (viz výčtové typy níže), úplný název typu `string` doplňující označení o hodnoty jeho parametrů a meze na ose x , tedy interval, kde se „něco děje“. Ty jsou nutné pro vykreslení charakteristických funkcí do grafu. Odhadoval jsem je na základě parametrů rozdělení.

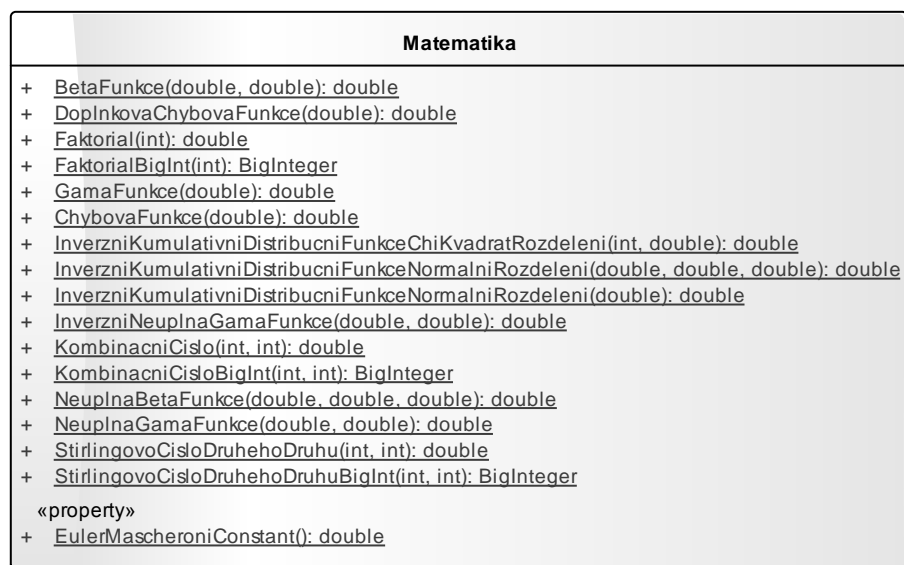
Každá třída kromě funkcí rozhraní obsahuje konstruktor, pomocí kterého jsou zadány parametry rozdělení. Jednotlivé výpočty charakteristik jsou implementovány dle předpisů ze zdrojů [9] a [14]. V několika případech je však bylo potřeba opravit. I zde jsem několik rozdělení doplnil například o parametr posunutí či možnost „roztažení“ na požadovaný interval. Strukturu naznačuje zjednodušený diagram tříd.



Obr. 43: Zjednodušený diagram tříd – charakteristiky náhodné proměnné

3.1.5 Matematika

Poslední částí vrstvy control je statická třída `Matematika`. Obsahuje potřebné matematické výpočty, které jsou zobrazeny na zjednodušeném diagramu třídy, kde jsou zachyceny pouze veřejné metody. Vzorce pro výpočet jednotlivých funkcí byly získány z literatury [11], [12] a [13].



Obr. 44: Zjednodušený diagram třídy Matematika

3.2 Entity

V této vrstvě jsou uchovány objekty, s kterými se dále pracuje. Důležitou roli zde hraje především histogram, který je nutný pro mnoho statistických výpočtů.

3.2.1 Histogram

Histogram zapouzdřuje kolekci jednotlivých tříd, které jsou uloženy v generickém seznamu `List<HistogramTrida>`, jednotlivé položky jsou tedy typu `HistogramTrida`. Samozřejmostí je implementace rozhraní `IEnumerable`, díky kterému může být použit i v cyklu `foreach`. Navíc je doplněn o `indexer`, takže je možné přistupovat k jednotlivým prvkům pomocí indexu. Mezi vlastnosti patří počet prvků představující součet četností jednotlivých tříd. Dále je zde uveden počet tříd a jsou poskytnuty i samotné třídy. Parametry a požadavky na histogram jsou zadávány pomocí konstruktoru.

Prvkem histogramu je tedy `HistogramTrida`. Hranice třídy a četnost prvků uvnitř jsou dány jako vlastnosti. Dále je přetížen operátor `++` tak, aby se při jeho použití zvýšila četnost třídy o jedna. To je potřeba při určování četností ze vstupní kolekce hodnot, na které se nyní podíváme.

Pro všechny prvky vstupní kolekce hodnot se vypočítá index, podle kterého jsou vkládány do histogramu. Nejsou v něm však ukládány konkrétní hodnoty, pouze je zvýšena četnost u příslušné třídy. Nyní si ukážeme vzorec pro získání indexu.

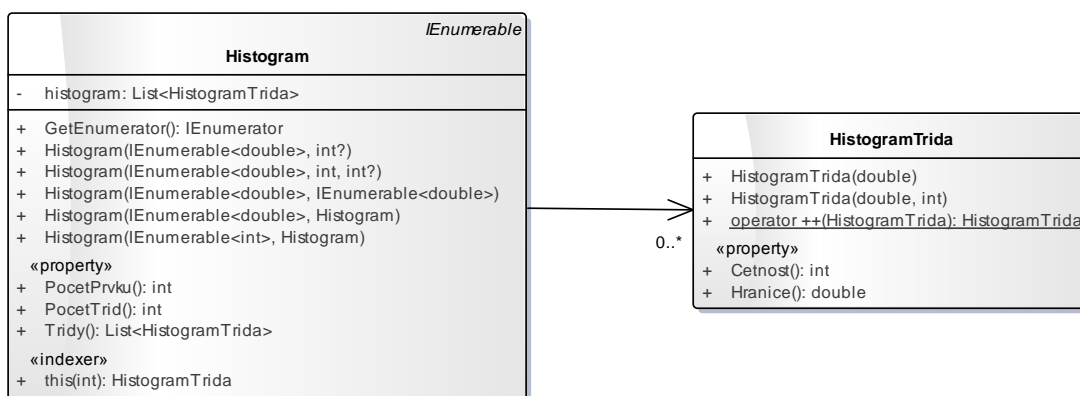
$$index = \frac{tridy(x - min)}{delka}, \quad delka \neq 0$$

$$= 0, \quad delka = 0$$

Proměnná x představuje aktuální prvek vstupní kolekce, hodnota $tridy$ udává počet tříd histogramu, min a $delka$ mají význam dle parametrů histogramu. Pokud je histogram vytvořen automaticky, min představuje minimální hodnotu a $delka$ pak rozdíl mezi maximální a minimální hodnotou, oba vztaženy ke vstupní kolekci. Zde musí být ošetřen případ, že by všechny vstupní hodnoty byly stejné, $delka$ by pak byla rovna zmíněné nule. V případě, že jsou třídy dány uživatelem, představuje min dolní mez první třídy a $delka$ rozdíl mezi horní mezí poslední třídy a hodnotou min . Po získání výsledného indexu je ještě třeba provést následující revizi.

$$\begin{aligned} index &= index - 1, & index &= [index] \\ index &= 0, & index &< 0 \\ index &= tridy - 1, & index &> tridy - 1 \end{aligned}$$

První podmínka má následující význam. Víme, že rozmezí hodnot třídy je dáno intervalem ($dolní\ mez$; $horní\ mez$). V případě, že $x = horní\ mez$, by byla hodnota x vložena do další (následující) třídy. Díky tomuto ošetření však bude vložena na správné místo. Hranaté závorky pak značí celou část desetinného čísla. Další dvě podmínky kontrolují, zda není hodnota indexu mimo rozsah histogramu. Tento stav může nastat, pokud uživatel nastaví třídy ručně a některá hodnota z kolekce vstupních dat je menší než dolní hranice první třídy, nebo větší než horní hranice poslední třídy. Zde je potřeba zmínit, že $index \in \{0; \dots; tridy - 1\}$. Po vyhodnocení podmínek je zvýšena četnost třídy na pozici $[index]$. I zde hranaté závorky značí celou část desetinného čísla.



Obr. 45: Zjednodušený diagram tříd – histogram

3.2.2 Výčtové typy

Implementované výčtové typy slouží k řízení běhu programu na základě potřeb uživatele a k identifikaci jednotlivých rozdělení pravděpodobnosti, konkrétně jeho názvu a typu. Navíc je minimalizováno opakování kódu, které by jinak vzniklo při výběru různých akcí, jako je třeba generování čísel, výpis charakteristik a podobně.



Obr. 46: Zjednodušený diagram tříd – výčtové typy

3.2.3 Nastavení

Část entity zakončíme statickou třídou, která v sobě uchovává různá nastavení aplikace. Jedná se například o zaokrouhlování, nastavení barev, grafů či histogramů. Výhledově by mohla být doplněna možnost ukládání uživatelského nastavení. Uživatel by pak měl po spuštění zachováno vlastní nastavení.

3.3 Presentation

Poslední vrstva slouží pro styk s uživatelem. Obsahuje jednotlivé formuláře a výstup hodnot z aplikace ve formě souborů.

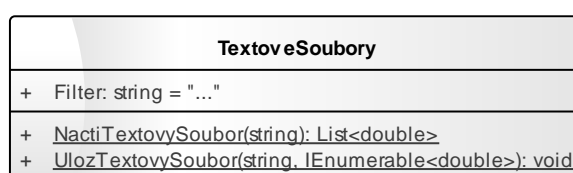
3.3.1 Formuláře

Ať už se jedná o hlavní formulář nebo dialogová okna pro nastavení aplikace, vždy slouží k zadání požadavků uživatele. Výstupem pak může být informace v textové či grafické podobě, nebo změna chování aplikace, například po změně nastavení grafu.

Jednotlivé formuláře jen volají metody tříd vrstvy control a využívají objekty z vrstvy entity. Takže například v případě chyby nějakého výpočtu stačí provést opravu pouze na jednom místě.

3.3.2 Textové soubory

K načítání a ukládání seznamů čísel slouží samostatná třída. V případě načítání dat pouze přečte řádky souboru a převede je na desetinné číslo. O to, zda splňují další požadavky, už se stará samotná logika aplikace. Pokud se jedná o ukládání dat, je parametrem pouze kolekce čísel. O správnost údajů se stará opět logika aplikace.



Obr. 47: Diagram třídy pro práci s textovými soubory

4 ZÁVĚR

Výsledná aplikace představuje nástroj, který vyniká nejen velkým množstvím funkcí, ale i velice příjemným, intuitivním a uživatelsky přívětivým grafickým rozhraním. Ve spojení se svižným chodem se pak stává vítaným pomocníkem na poli pracovním i studijním. Určena je především pro odborníky v oblasti spojitě či diskrétní simulace a vysokoškolské studenty zabývající se problematikou vyžadující znalost a použití teorie pravděpodobnosti či matematické statistiky. Své uplatnění najde i v mnoha jiných oborech.

Součástí projektu je i publikace obsahující potřebný teoretický aparát, podrobný manuál k ovládání aplikace a stručný popis její vnitřní struktury. Pro využívání nástroje tudíž není nutná žádná další literatura a díky přítomnosti návodu si uživatel velice rychle osvojí práci s programem a dokáže tak ve velmi krátké době získat požadované informace.

Validace celého díla byla provedena ve dvou krocích. V prvním stádiu došlo k porovnání výstupů aplikace s odbornou literaturou. Ve druhé fázi bylo její chování posouzeno kvalifikovaným expertem z příslušné oblasti. Ten zároveň provedl kontrolu teoretické části publikace. Měla by tak být zajištěna správnost jednotlivých výstupů programu i přiloženého odborného textu.

Mohu tedy uvést, že zadání diplomové práce bylo splněno v plném rozsahu. Práce byla odevzdána v řádném termínu. Veškeré použité podklady a literární prameny jsou uvedeny v seznamu použité literatury.

Aplikace by i přes svůj rozsah mohla být dále rozšířena. Nabízelo by se hned několik možností. První a asi i nejnáročnější by bylo automatické rozpoznání rozdělení pravděpodobnosti při analýze dat. Dále by se dalo využít dnešních moderních technologií a pro zrychlení běhu implementovat paralelní zpracování výpočtů vícejádrovými procesory. Nakonec by bylo vhodné lokalizovat grafické prostředí do dalších jazykových mutací.

5 POUŽITÁ LITERATURA

1. NADRCHAL, Tomáš. 2013. *Generátory pseudonáhodných čísel rozdělení pravděpodobností*. Pardubice. Bakalářská práce. Univerzita Pardubice, Dopravní fakulta Jana Pernera.
2. HUŠEK, Roman a Josef LAUBER. *Simulační modely*. Praha: SNTL - Nakladatelství technické literatury, 1987. DT 330.116.1(075.8).
3. KNUTH, Donald Ervin. 2010. *Umění programování: 2. díl - Seminumerické algoritmy*. Vyd. 1. Brno: Computer Press, xi, 763 s. ISBN 978-80-251-2898-5.
4. RYBA, Bronislav. 2012. *Generování pseudonáhodných čísel*. Brno. Bakalářská práce. Mendelova univerzita v Brně, Provozně ekonomická fakulta.
5. KAVIČKA, Antonín. *Modelování a simulace* [intranet]. Pardubice, 2012 [cit. 2015-05-10]. Dostupné z: <http://fei-learn.upceucebny.cz>
6. BANKS, Jerry. 1998. *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*. Norcross, Ga.: Wiley-Interscience, xii, 849 p. ISBN 04-711-3403-1.
7. ZÁHOROVÁ, Věra. *Testy shody* [intranet]. Pardubice, 2011 [cit. 2013-05-19]. Dostupné z: <https://portal.upce.cz/portal/moje-studium/materialy.html>
8. Runs Up, Runs Down. 2000. *State University of New York At Oswego* [online]. [cit. 2015-05-10]. Dostupné z: <http://www.oswego.edu/~lwahl/classes/csc454/site/runsWithMean.html>
9. LINDA, Bohdan. *Pravděpodobnost*. Vyd. 1. Pardubice: Univerzita Pardubice, Fakulta ekonomicko-správní, 2010, 167 s. ISBN 978-80-7395-303-4.
10. KAHOUNOVÁ, Jana. 2008. Asymptotické pravděpodobnostní rozdělení výběrového maxima. In: *Acta Oeconomica Pragensia: Vědecký časopis Vysoké školy ekonomické v Praze* [online]. [cit. 2015-05-14]. Dostupné z: <https://www.vse.cz/polek/download.php?jnl=aop&pdf=103.pdf>
11. KNUTH, Donald Ervin. 2008. *Umění programování: Základní algoritmy*. Vyd. 1. Brno: Computer Press, xix, 648 s. ISBN 978-80-251-2025-5.
12. Stirling numbers of the second kind - MATLAB. 2015. *MathWorks - MATLAB and Simulink for Technical Computing* [online]. [cit. 2015-05-10]. Dostupné z: http://www.mathworks.com/help/symbolic/mupad_ref/combinat-stirling2.html

13. PRESS, William H., Saul A. TEUKOLSKY, William T. VETTERLING a Brian P. FLANNERY. *Numerical Recipes: The Art of Scientific Computing*. 3rd ed. Cambridge: Cambridge University Press, 2007, xxi, 1235 s. ISBN 978-0-521-88068-8.
14. SAUCIER, Richard. *Computer Generation of Statistical Distributions* [online]. 2000 [cit. 2013-05-19]. ISBN [9781423540281]. Dostupné z: <http://ftp.arl.mil/random/random.pdf>

6 PŘÍLOHY

PŘÍLOHA A <i>Přiložené datové médium</i>	73
--	----

PŘÍLOHA A *Přiložené datové médium*

K diplomové práci je přiloženo datové médium se zkušební aplikací a kopií této práce ve formátu *.pdf.