UNIVERZITA KONŠTANTÍNA FILOZOFA V  NITRE

FAKULTA PRÍRODNÝCH VIED A  INFORMATIKY

**Umelá inteligencia s  využitím Pythonu**

BAKALÁRSKA práca

**2022 Johana Heneková**

UNIVERZITA KONŠTANTÍNA FILOZOFA V  NITRE

FAKULTA PRÍRODNÝCH VIED A  INFORMATIKY

**Umelá inteligencia s  využitím Pythonu**

BAKALÁRSKA práca

Študijný odbor: 18. Informatika

Študijný program: Aplikovaná informatika

Školiace pracovisko: Katedra informatiky

Školiteľ: Mgr. Ľubomír Benko, Ph.D .

Nitra 2022 Johana Heneková



Univerzita Konštantína Filozofa v  Nitre

Fakulta prírodných vied a  informatiky

**ZADANIE ZÁVEREČNEJ PRÁCE**

|  |  |  |
| --- | --- | --- |
| **Meno a  priezvisko študenta:** | | Johana Heneková |
| **Študijný program:** | | aplikovaná informatika (Jednoodborové štúdium, bakalársky I . st., denná forma) |
| **Študijný odbor:** | | informatika |
| **Typ záverečnej práce:** | | Bakalárska práca |
| **Jazyk záverečnej práce:** | | slovenský |
| **Sekundárny jazyk:** | | anglický |
| **Názov:** | Umelá inteligencia s  využitím Pythonu | | |
| **Anotácia:** | V  dnešnej dobe sa s  rastúcim množstvom a  zložitosťou dát snaží vyrovnať umelá inteligencia a  strojové učenie. Presnejšie predpovede a  získané znalosti pomocou umelej inteligencie zlepšujú efektivitu priemyslu, znižujú výrobné náklady a  zvyšujú produktivitu. Práve kvôli veľkému množstvu dát, musí umelá inteligencia byť schopná spracovať toto množstvo dát efektívne a  v  časovo prijateľných hraniciach. Z  toho dôvodu sa stále častejšie využíva pri umelej inteligencii jazyk Python, keďže jeho syntax nie je príliš komplexná a  dokáže sa vysporiadať so zložitými procesmi v  riešenej problematike.  Cieľom práce je vytvoriť sériu aplikácií umelej inteligencie riešenej pomocou jazyka Python.  Charakter práce: aplikačný.  Požiadavky na obsah podľa charakteru práce (nemyslí sa tým šablóna, tá je daná bez ohľadu na charakter práce): popis riešeného problému, návrh systému/hardvérového riešenia a  pod. (modely, ..), metodika vývoja/tvorby, implementácia, popis vytvoreného riešenia, testovanie.  Požiadavky na vedomosti a  zručnosti študenta: dobrá znalosť anglického jazyka, dobrá znalosť z  programovania, analytické myslenie. | | |
| **Školiteľ:** | Mgr. Ľubomír Benko, Ph.D . | | |
| **Oponent:** | RNDr. Ján Skalka, PhD. | | |
| **Katedra:** | KI - Katedra informatiky | | |

**Dátum zadania:** 27.10.2020

**Dátum schválenia:** 05.04.2021 RNDr. Ján Skalka, PhD., v . r .

vedúci/a  katedry

# Poďakovanie

Veľmi rada by som sa chcela poďakovať svojmu školiteľovi, pánovi Mgr. Ľubomírovi Benkovi, Ph.D . za odborné rady, pomoc pri práci, spoluprácu a  hlavne za ochotu. Taktiež sa chcem poďakovať za spoluprácu a  konštruktívnu kritiku od študentov predmetu Programovacie jazyky pre umelú inteligenciu (PJUI) v  roku 2022.

# ABSTRAKT

HENEKOVÁ, Johana: Umelá inteligencia s  využitím Pythonu. [Bakalárska práca]. Univerzita Konštantína Filozofa v  Nitre. Fakulta prírodných vied a  informatiky. Školiteľ: Mgr. Ľubomír Benko, Ph.D . Stupeň odbornej kvalifikácie: Bakalár odboru Aplikovaná informatika. Nitra: FPVaI, 2022. 49 s .

Bakalárska práca sa zaoberá problematikou vytvárania kurzu pre výučbu strojového učenia. Kurz je vytvorený za pomoci programovacích jazykov Python a  Markdown a  pri práci využíva knižnice Scikit-learn, Python Pandas, Seaborn a  NumPy. Práca je rozdelená do piatich kapitol. Teoretická časť sa zaoberá hlavne popísaním a  vysvetlením umelej inteligencie a  strojového učenia. Ďalej popisuje aj dopyt po kurze, ktorý by sa venoval strojovému učeniu na Univerzite Konštantína Filozofa v  Nitre. Ďalšia časť práce popisuje samostatný návrh kurzu, kde sú zanalyzované dostupné knižnice a  nástroje využívané v  oblasti strojového učenia a  stránka Priscilla, ktorá bude prostriedkom na distribúciu kurzu. Štvrtá kapitola popisuje samotný postup vytvárania kurzu a  posledná kapitola hodnotí výsledky, ktoré boli vytvorené analýzou dotazníka. Dotazník bol distribuovaný vzorke študentov, ktorá otestovala finálnu podobu kurzu a  spravila spätnú väzbu. V  závere sa nachádzajú odporúčania a  výsledky práce.

Kľúčové slová: Strojové učenie. Python. Kurz. Umelá inteligencia. Markdown

# ABSTRACT

HENEKOVÁ, Johana: Artificial Intelligence with Python. [Bachelor Thesis]. Constantine the Philosopher University in Nitra. Faculty of Natural Sciences and Informatics. Supervisor: Mgr. Ľubomír Benko, Ph.D . Degree of Qualification: Bachelor of Applied Informatics. Nitra: FNSaI, 2022 49 p .

The bachelor thesis deals with the issue of creating a  course for teaching machine learning. The course is created with the help of programming languages Python and Markdown and uses the libraries Scikit-learn, Python Pandas, Seaborn and NumPy. The work is divided into five chapters. The theoretical part deals mainly with the description and explanation of artificial intelligence and machine learning. It also describes the demand for a  course that would focus on machine learning at the University of Constantine the Philosopher in Nitra. The next part of the work describes a  separate course design, which analyzes the available libraries and tools used in the field of machine learning and the Priscilla site, which will be a  means of distributing the course. The fourth chapter describes the actual course of creating the course and the last chapter evaluates the results that were created by analyzing the questionnaire. The questionnaire was distributed to a  sample of students who tested the final form of the course and provided feedback. In the end, there are recommendations and results of the work.

Keywords: Machine learning. Python. Course. Artificial intelligence. Markdown.

# Obsah

[Úvod ..................................................................................................................................9](#_Toc100674457)

[1 Analýza súčasného stavu 10](#_Toc100674458)

[1.1  Čo je to umelá inteligencia 10](#_Toc100674459)

[1.2  Momentálne využitie umelej inteligencie 11](#_Toc100674460)

[1.3  Budúcnosť s  umelou inteligenciou 12](#_Toc100674461)

[1.4  Strojové učenie 13](#_Toc100674462)

[1.5  Využitie v  záverečnej práci 14](#_Toc100674463)

[1.6  Využitie v  univerzitnej sfére 14](#_Toc100674464)

[1.7  Kurzy vo svete 14](#_Toc100674465)

[1.8  Programovací jazyk Python 16](#_Toc100674466)

[2  Ciele záverečnej práce 19](#_Toc100674467)

[3 Návrh kurzu 20](#_Toc100674468)

[3.1  Knižnice 20](#_Toc100674469)

[3.1.1  Scikit-learn 20](#_Toc100674470)

[3.1.2  TensorFlow 20](#_Toc100674471)

[3.1.3  XGBoost 21](#_Toc100674472)

[3.1.4  Keras Python 21](#_Toc100674473)

[3.1.5  NumPy Python 21](#_Toc100674474)

[3.1.6  Theano Python a  PyTorch Python 22](#_Toc100674475)

[3.1.7  Python Pandas 22](#_Toc100674476)

[3.1.8  Seaborn Python 23](#_Toc100674477)

[3.1.9  Zhrnutie 23](#_Toc100674478)

[3.2  Nástroje 24](#_Toc100674479)

[3.2.1  Anaconda 24](#_Toc100674480)

[3.2.2  Jupyter Lab 24](#_Toc100674481)

[3.3  Priscilla 25](#_Toc100674482)

[3.4  Kurz 26](#_Toc100674483)

[4 Postup vytvárania kurzu 27](#_Toc100674484)

[4.1  Využitie knižníc 27](#_Toc100674485)

[4.2  Proces výberu tém pre kurz 29](#_Toc100674486)

[4.2.1  Úvod do kurzu 30](#_Toc100674487)

[4.2.2  Knižnice 30](#_Toc100674488)

[4.2.3  Iris dataset 30](#_Toc100674489)

[4.2.4  Predspracovanie dát 31](#_Toc100674490)

[4.2.5  Učenie s  učiteľom 32](#_Toc100674491)

[4.2.6  KNN 33](#_Toc100674492)

[4.2.7  Lineárna regresia 33](#_Toc100674493)

[4.2.8  Porovnávanie modelov a  trénovacie a  testovacie dáta 34](#_Toc100674494)

[4.2.9  Záver kurzu 34](#_Toc100674495)

[5 Výsledky riešenia a  ich zhodnotenie 35](#_Toc100674496)

[5.1  Práca v  Jupyter notebookoch 35](#_Toc100674497)

[5.2  Dotazník a  kurz v  praxi 35](#_Toc100674498)

[5.2.1  Vzorka študentov 36](#_Toc100674499)

[5.2.2  Úvodná kapitola a  knižnice 36](#_Toc100674500)

[5.2.3  Zrozumiteľnosť 38](#_Toc100674501)

[5.2.4  Náročnosť 39](#_Toc100674502)

[5.2.5  Množstvo úloh 40](#_Toc100674503)

[5.2.6  Množstvo teórie 41](#_Toc100674504)

[5.2.7  Po vypracovaní kurzu 42](#_Toc100674505)

[5.2.8  Spätná väzba 44](#_Toc100674506)

[5.2.9  Záver dotazníka 45](#_Toc100674507)

[Záver 46](#_Toc100674508)

[Zoznam bibliografických odkazov 47](#_Toc100674509)

[Zoznam príloh 51](#_Toc100674510)

# Úvod

Umelá inteligencia sa objavuje čoraz viac v  našom živote. Ako sa rozširuje jej využitie, tak by sa malo rozširovať aj vzdelanie o  nej. Táto myšlienka bola kľúčová pri tvorbe našej práce, keďže našim cieľom bolo vytvorenie kurzu, ktorý by sa zaoberal úvodom do umelej inteligencie.

Vychádzali sme z  momentálnej ponuky kurzov na stránke Priscilla, kde sme chceli vyplniť dopyt po úvode do umelej inteligencie. Z  tohto dôvodu sme sa nezaoberali ani celou tematikou umelej inteligencie, ale konkrétne strojovým učením, aby sme v  praxi splnili význam slova úvod.

Naše ciele boli hlavne aby bol kurz jednoduchý, ale zároveň vysvetlil základné témy strojového učenia a  najmä aby podnietil záujem o  túto tému. Z  tohto dôvodu sme nešli do hĺbky, ale podporili sme aj túžbu po hlbšom vzdelaní odporúčanými stránkami na konci kapitol.

V  práci sme kládli dôraz taktiež na praktické využitie. Síce sme nerozoberali komplikované tematiky s  komplexnými úlohami, ale navrhli sme skôr jednoduché praktické zadania na programovanie v  jazyku Python a  taktiež do kurzu integrovali teoretické otázky na interaktívnosť v  kurze.

Kurz bol vytváraný postupne za pomoci analýzy výučbových materiálov iných univerzít a  to s  cieľom poskytnúť dostatočné informácie a  naučiť študentov novým poznatkom.

Finálnu podobu kurzu mali možnosť otestovať študenti na našej univerzite a  za pomoci ich spätnej väzby sme poukázali na nedostatky, navrhli sme odporúčania a  vyzdvihli sme kladné stránky kurzu. Tieto poznatky sa môžu zužitkovať v  budúcnosti, pri dodatočných úpravách.

# 1 Analýza súčasného stavu

Táto kapitola predstavuje umelú inteligenciu, kde sa momentálne využíva, prednesie jej možnú budúcnosť a  problémy ktoré sa s  ňou spájajú. V  neskorších kapitolách popisuje strojové učenie, ktorému sme sa venovali v  našom kurze a  porovná výučbu strojového učenia na viacerých univerzitách.

## 1.1  Čo je to umelá inteligencia

S  umelou inteligenciou (artificial intelligence, AI) sa stretávame v  dnešnej dobe vo veľa médiách a  často krát počúvame o  jej dopadoch na náš život. Preto považujeme za dôležité najskôr objasniť, čo si môžeme predstaviť pod umelou inteligenciou.

John McCarthy popísal tento pojem vo svojej knihe nasledovne: Umelá inteligencia je vedecký a  inžiniersky postup pri tvorbe inteligentných strojov, obzvlášť pri tvorbe inteligentných počítačových programov.(McCarthy, 2007)

Iný pohľad priniesol Patrick D . Smith keď umelú inteligenciu popísal takto: Umelá inteligencia je systém, ktorý dovoľuje počítačom aby vykonávali úlohy bez dopredu presného naprogramovania.(Smith, 2018)

Tieto pohľady sa líšia najmä v  tom, že jeden opisoval tvorbu už priamo strojov a  druhý opisoval ako daný systém funguje. Je to hlavne kvôli tomu, že je umelá inteligencia ešte stále iba vo vývoji.

Síce začiatky modernej AI by sa dali pripísať McCarthymu, ale za najznámejšieho sa pokladá Alan Turing. Turing napísal v  roku 1950 veľmi populárny článok (Turing, 1950) o  filozofickej stránke umelej inteligencie, kde taktiež opísal okrem teórie výpočtov aj dopad aký by mala. Už vtedy, pri počiatočných myšlienkach o  základoch umelej inteligencie opisoval inteligenciu strojov, ktorá by bola rovnaká ako ľudská. Čo je veľmi dôležité, pretože ľudská inteligencia bola vždy cieľom a  akýmsi vzorom pre úplnú funkcionalitu umelej inteligencie. Dá sa teda povedať, že sa vývojári vždy snažili spraviť robota s  mysľou človeka, ktorý by nám slúžil.(Patterson, 2002)

Nápady na takéhoto robota boli už pred vznikom informatiky a  počítačov. Boli to pokusy o  načrtnutie základov pre inteligentné stroje. Napríklad už Homér opísal automaty v  jeho epickej básni Illiada v  ôsmom storočí pred Kristom. Postupne sa tieto predstavy vyvíjali až po skonštruované stroje, ktorých jediný zmysel bolo hrať šach. (McCorduck, Cfe, 2004)

V  dnešnej dobe má umelá inteligencia veľa oblastí a  tie sa zaoberajú mnohými problematikami, ktoré sú samé o  sebe veľmi komplexné. Jednou z  nich sú napríklad neurónové siete, ktoré si zachovali do určitej miery pripodobnenie s  ľudskou inteligenciou a  pracujú s  modelom neurónov.(Mijwel, 2018)

Taktiež spomenieme expertné systémy, ktoré sú založené na databázach a  sériách pravidiel s  rozhodovaním ak/potom. (Patterson, 2002)

Umelá inteligencia je oblasť vedy, ktorá vytvára inteligentné stroje a  počítačové programy na vykonávanie rôznych úloh, ktoré vyžadujú prítomnosť ľudskej inteligencie.(Dick, 2019)

## 1.2  Momentálne využitie umelej inteligencie

Umelá inteligencia v  dnešnom svete neustále napreduje s  novými inováciami. Dnešné počítačové systémy sú navrhnuté tak, aby vykonávali pre nás malé a  často krát aj automatické úlohy ako napríklad rozpoznávanie tvárí, šoférovanie áut alebo riešenie rôznych matematických úloh. Avšak hlavnou úlohou umelej inteligencie je vyvinúť viac pokročilé a  komplexné systémy, ktoré by eventuálne prekonali ľudí. Príkladom takýchto komplexných úloh je aj hranie šachu. Pri hraní treba premýšľať niekoľko ťahov dopredu či už vlastných alebo protivníkových. Treba taktiež uvažovať o  všetkých možných vývojoch hry a  tieto výpočty opakovať po každom ťahu. Preto výhra počítača nad majstrom sveta v  šachu bola veľkým míľnikom v  histórii umelej inteligencie. Tento historický moment sa stal v  roku 1997, kedy počítač Deep Blue porazil hráča Garryho Kasparova.(Hassabis, 2017)

Dnešná doba už počíta s  umelou inteligenciou vo veľa smeroch a  pomaly sa rozširuje do ďalších. Napríklad čoraz viac využívame na cestách automaticky riadené autá. Dá sa povedať, že vo veľa prípadoch sedí v  aute človek na to, aby riešil nepredvídateľné stavy. Čiže predstavuje akéhosi kontrolóra, ktorý pozoruje či ide automatický systém v  poriadku. Samozrejme je táto sféra umelej inteligencie stále vo vývoji a  nevieme sa na ňu spoľahnúť dostatočne na to, aby sme jej zverili ľudské životy na cestách.(Hong a  kol., 2021)

Ďalšou veľmi zaujímavou a  populárnou sférou je robotika. Roboty s  umelou inteligenciou využívame napríklad na veľmi užitočné práce a  to v  prípadoch, kedy by bol ohrozený ľudský život. Napríklad zdvíhanie ťažkých vecí, práca so žeravými kovmi alebo aj práca vo fabrikách. Tam sa vyžaduje vykonávať rovnakú činnosť dokola a  ľudská psychika by to niesla zle, dokonca s  možnými trvalými následkami a  poruchami mozgu.(Owen-Hill, 2021)

Umelá inteligencia je aj pri tak malých veciach ako napríklad GPS, kedy sa nám prepočítava neustále najlepšia trasa, ale aj pri tak obrovských ako sú chirurgické operácie.(Poola, 2017)

Umelá inteligencia je prítomná pri veľa každodenných činnostiach a  postupne sa dostáva do širšieho diania s  väčšími možnosťami a  právami.

## 1.3  Budúcnosť s  umelou inteligenciou

Budúcnosť umelej inteligencie je prekonať vo všetkých smeroch ľudí a  priniesť výhodnejšie riešenia a  teda šetriť čas. Avšak tento plán a  vízia prináša aj obavy. Ak sa umelej inteligencii prenechá každodenné rozhodovanie ľudí v  bežných situáciách, môže nastať problém odosobnenia sa od vlastných životov a  ich riadenia. Je veľké množstvo filmov, kde vidíme tento efekt, keď hlavný hrdina iba „pláva životom“ a  nemusí o  ničom rozhodovať a  iba si ho užívať, ale taktiež vidíme ako sa stráca radosť a  potešenie zo samotného žitia. To je jedna z  mnohých filmármi predostretých scenárov. Tie ostatné, kde napríklad umelá inteligencia obíde Asimovove zákony robotiky a  zničí ľudstvo sú horšie. Teraz poukážeme na príklady reálnejších a  aj momentálne často diskutovaných problémov prepojených s  umelou inteligenciou.

Veľa ľudí už teraz namiesto premýšľania okamžite siaha po vyhľadávacích službách prostredníctvom internetu aby dostali odpoveď na svoju, často krát, jednoduchú otázku a  teda nemuseli premýšľať navyše. Už existuje aj veľa štúdií, ktoré upozorňujú na tento efekt „Googlenia“ a  aké problémy vie spôsobiť v  budúcnosti. Napríklad sa hovorí o  zníženej pozornosti v  dôsledku toho, že máme okamžité odpovede, zníženie výkonu u  kognitívnych úloh alebo aj zhoršenie v  sociálnych zručnostiach.(Wilmer a  kol., 2017)

Ďalším z  veľkých problémov, ktoré sa už ale do veľkej miery snažíme eliminovať je zaobchádzanie s  osobnými údajmi. Pri ďalšom vývoji AI bude táto otázka ešte viac zložitejšia na zodpovedanie a  hranica našich osobných údajov sa bude pravdepodobne posúvať.(Etzioni, Etzioni, 2016) Efekt „Googlenia“ sa prejavuje aj tu, lebo častokrát veríme, že online svet je nejaká pamäťová schránka, do ktorej vieme vložiť svoje údaje a  informácie bez následkov s  pocitom bezpečia. Jedno z  riešení je napríklad možnosť podania žiadosti o  zmazanie informácií zo služby Google[[1]](#footnote-1).

Aj keby sme si dávali pozor na to, čo zadávame na internet, tak veľa ľudí sa obáva o  svoju prácu. V  predošlej kapitole sme poukázali na to, že veľa robotov vykonáva monotónne práce, čo je len malé percento oblastí, kde umelá inteligencia nahrádza ľudskú prácu.(Su, 2018)

V  úvode sme spomenuli ublíženie umelej inteligencie ľuďom v  prípade, ak AI obíde Asimovove zákony robotiky. Avšak umelá inteligencia nemusí byť ani na takej úrovni, aby ublížila. Stačí úplne prípad, kedy ju jedna skupina ľudí zneužije a  za jej pomoci uškodí druhej skupine. Ako príklad uvedieme umelú inteligenciu v  armádnej sfére. Tam sa môže stať to, že by sme AI dali cieľ, ktorý by nasledovala, avšak zlým spôsobom s  vážnymi následkami na životoch. (Anderson, 1989)

Umelá inteligencia sa považuje za veľmi rýchlo rastúcu oblasť informatiky. Nie je to iba po vývojovej stránke ale aj výpočtovej. Už v  roku 2019 vedci zistili, že umelá inteligencia nenasleduje Moorove pravidlo. Toto pravidlo hovorí o  predpokladanom náraste výpočtovej rýchlosti procesorov. Momentálne sa odhaduje, že sa zdvojnásobí každých 18 mesiacov. Avšak pri umelej inteligencii je to odhadované iba na 3  mesiace. (Saran, 2019)

## 1.4  Strojové učenie

Pri strojovom učení (machine learning) učíme počítač alebo program využívať svoje vlastné poznatky z  minulosti tak, aby mal lepšie výsledky v  budúcnosti. (Anderson a  kol., 2018)

Je to odvetvie umelej inteligencie a  informatiky, ktoré sa zameriava na používanie údajov a  algoritmov na nasimulovanie spôsobu, akým sa ľudia učia a  tým postupne zlepšovať presnosť budúcich výsledkov.(IBM, 2020)

Príkladom môže byť program, ktorý má filtrovať spam (nevyžiadaná pošta) v  poštovej komunikácii. Je veľa prístupov ako riešiť túto problematiku, ale najjednoduchšie je naučiť počítač, aby to robil sám. To znamená, že postupným učením počítač kategorizuje emailovú komunikáciu na spam a  non-spam (vyžiadaná pošta). Robí to na základe podobností a  postupne sa v  tejto činnosti zdokonaľuje ako aj človek, ktorý si postupne dáva pozor a  rozoznáva nevhodný mail.(Anderson a  kol., 2018)

## 1.5  Využitie v  záverečnej práci

V  našej práci sme pracovali so strojovým učením. Túto oblasť sme si vybrali z  dôvodu, že je začiatočnícky prijateľná a  teda spĺňa náš cieľ spraviť úvodný kurz. Dobre demonštruje veľa problematík typických pre AI, predstavuje viacero modelov a  vo výsledku tak vieme na nej ukázať základné princípy fungovania umelej inteligencie. Zároveň sme brali do úvahy aj obtiažnosť a  komplexnosť, ktoré sme vedeli regulovať a  teda sme tým nechceli zahltiť študenta priveľa zložitými témami, ale dať mu skôr ľahký náhľad do problematík. Strojové učenie má prístupných veľa či už knižných materiálov alebo priamo kurzov na to, aby študent mohol pokračovať po ukončení nášho kurzu v  študovaní problematiky a  my sme mali dostatočný rozhľad na vytvorenie kurzu.

## 1.6  Využitie v  univerzitnej sfére

Na univerzite Konštantína Filozofa v  Nitre je na bakalárskom štúdiu informatiky povinný predmet o  Umelej inteligencii[[2]](#footnote-2). Tento predmet je zameraný na zoznámenie študentov so základnými problémami a  teóriou umelej inteligencie. Na cvičeniach sa pracuje v  programovacom jazyku Prolog[[3]](#footnote-3), na ktorom sa opisujú základné princípy logických úvah a  ako ich vie počítač spracovať.

Náš kurz teda môže byť dobrým prostriedkom na prehĺbenie si znalostí po absolvovaní tohto predmetu ako aj doplnkovým materiálom pri štúdiu umelej inteligencie. Kurz sa bude v  budúcnosti nachádzať na stránke Priscilla[[4]](#footnote-4), čiže bude možné absolvovať kurz kedykoľvek a  s  automatickou spätnou väzbou. Stránka je aj voľne prístupná pre verejnosť, čiže má možnosť nárastu používateľov aj mimo akademickej sféry. Nástroj Priscilla predstavíme bližšie neskôr v  kapitole 3 .3 .

## 1.7  Kurzy vo svete

Kurzy, kde sa vyučuje strojové učenie, sú na informatických oboroch na vysokých školách veľmi časté. Považuje sa to za základnú programátorskú zručnosť a  prinášajú často krát nový pohľad a  nový spôsob zmýšľania sa nad problematikami aj z  iných sfér.

Ako prvé sme pozerali obsahy kurzov zo škôl z  okolia. Masarykova univerzita v  Brne ponúka okrem štúdia priamo na Katedre strojového učenia a  spracovania dát[[5]](#footnote-5) aj viacero predmetov z  tejto oblasti. Predmety, ktoré sú vyučované rôznymi spôsobmi a  v  rôznych programovacích jazykoch. Napríklad jeden je cielený na programovanie v  jazyku R [[6]](#footnote-6). Tento jazyk využívajú viaceré univerzity na programovanie umelej inteligencie, avšak nám neprišiel najvhodnejší vzhľadom na jeho náročnejšiu syntax a  vo všeobecnosti menej intuitívne programovanie, kde Python rozhodne vyniká. Viacero predmetov, ktoré sme našli, bolo vedených najmä teoreticky[[7]](#footnote-7) alebo s  minimom programovania.

V  ponuke na Masarykovej univerzite je aj predmet, kde sa vyučuje strojové učenie v  programovacom jazyku Python[[8]](#footnote-8). Predmet sa zaoberá najskôr históriou AI aj strojovým učením, potom prechádza do praktickej časti ukazovaním modelov a  hypotéz, neskôr predstavuje regresiu a  rozhodovacie stromy. Tieto časti sme chceli obsiahnuť v  našom kurze, avšak náš kurz nemohol pokrývať dĺžku semestra a  teda sme sa rozhodli vynechať históriu umelej inteligencie. Taktiež to bolo z  dôvodu, že sme chceli kurz smerovať praktickým smerom už od začiatku a  históriu sme skôr brali ako dodatočný zdroj na štúdium. Ďalej sa predmet avšak zaoberá aj neurónovými sieťami, čo bolo taktiež mimo smerovania nášho kurzu, keďže ide o  náročnú tému, ktorú sme skôr navrhovali na samostatný kurz.

Zvyšné pozorované univerzity, mali menšie možnosti na štúdium strojového učenia. Napríklad sme analyzovali predmet na Vysokom učení technickom v  Brne, ktorý vyučuje umelú inteligenciu[[9]](#footnote-9), avšak z  jeho anotácie sme nezistili, či zahŕňa aj praktické programovanie a  ak áno nezistili sme v  akom jazyku. Obsahom bol kurz podobný spomínaním kurzom na Masarykovej univerzite.

Na Technickej univerzite v  Košiciach sa predmet o  inteligentných systémoch taktiež vyučuje v  jazyku Python. Okrem tejto praktickej časti sa predmet zameriava na približne rovnaké témy ako predošlé kurzy, vo všeobecnosti ale smeruje do väčšej hĺbky ako by sme potrebovali. Medzi kapitolami nás upútala téma o   predspracovaní dát, ktorú sme zvažovali na pridanie do nášho kurzu.

Taktiež sme vybrali témy ako zhlukovanie (clustering), kanály (pipeline) alebo vyhľadávacie algoritmy, ktoré tvorili našu rezervu. Tieto kapitoly sme chceli zahrnúť v  prípade, že by sme potrebovali navýšiť obsah kurzu alebo by sme ich potrebovali opísať z  dôvodu logickej nadväznosti.

V  analýze kurzov sme okrem vyššie popísaných obsahov ktoré sa vyučujú najčastejšie teoreticky alebo kombinovane- programovanie a  teória, a  najčastejšie v  jazykoch R  a  Python.

Toto zistenie podporilo náš výber jazyka. Zároveň sme z  tejto analýzy zhodnotili, čo by mal správny kurz obsahovať a  aké témy sme mali zahrnúť určite. Vo všeobecnosti sme našli podobnosť v  téme o  učení s  učiteľom a  bez učiteľa, o  rozhodovacích stromoch a  rozoberaní rôznych modelov. Tieto kapitoly nám prišli vhodné ako úvod do strojového učenia, keďže boli aj často vyučované na predmetoch univerzít.

## 1.8  Programovací jazyk Python

Programovací jazyk Python[[10]](#footnote-10) je interpretovaný, objektovo orientovaný, vysoko úrovňový jazyk s  dynamickou sémantikou.(Ceder, 2018)

Je obľúbenou voľbou programátorov vďaka tomu, že sa nemusí kompilovať a  teda aj ladenie kódu je jednoduchšie. Zároveň má Python veľké množstvo knižníc, ktoré sa dajú využiť pre ešte rýchlejšie a  efektívnejšie programovanie.(Ceder, 2018)

Pri výbere sme najskôr spravili analýzu toho, ktoré programovacie jazyky sa momentálne využívajú.

Chart, bar chart

Description automatically generated

Obrázok 1 Porovnanie programovacích jazykov, august 2021[[11]](#footnote-11)

Na obrázku je znázornená popularita používania jazyka Python k  mesiacu august v  roku 2021. Programovacie jazyky sú porovnávané v  dvoch kategóriách: percentuálne hľadanie v  oblasti PYPL (PopularitY of Programming Language), ktorá vznikla analýzou vyhľadávania tutoriálov v  danom jazyku na vyhľadávacom engine Google. Druhé hľadanie je kategória TIOBE udaná v  percentách. Ide o  množstvo hľadaní daného programovacieho jazyka na vzorke 25 vyhľadávacích prostriedkov (search engine). Z  grafu vieme vypozorovať, že Python absolútne dominuje v  kategórii PYPL so skoro 30% a  v  kategórii TIOBE je druhý po C /C ++.

Ďalšia, pre nás dôležitá informácia pri vyberaní jazyka bola, či je Python vhodný na výučbu. Keďže nie vždy musí byť najpoužívanejší jazyk aj najvhodnejší na učenie sa novej oblasti informatiky. Môže mať napríklad komplikovanú syntax a  priveľa možností ako zdokonaľovať kód čo by viedlo k  náročnejšiemu vysvetľovaniu už len programátorských princípov.

Pre nás bol Python jednoznačne najvhodnejšia voľba, má jednoduchú syntax, ktorá sa ľahko učí a  je zrozumiteľná, má veľké množstvo knižníc, ktoré boli pre náš kurz vyhovujúce a  je aj aktuálny vo sfére programovania a  učenia. Pre hlbšiu analýzu odporúčame článok na stránke Springboard[[12]](#footnote-12), kde autorka Sakshi Gupta popisuje rozdiely a  používanie jednotlivých programovacích jazykov.

Na základe uvedených dôvodov sme vybrali Python ako hlavný programovací jazyk pre tvorbu kurzu. Zároveň Python aj po tridsiatich rokoch a  veľkej konkurencii iných programovacích jazykov stále dominuje vo veľa smeroch a  predpokladáme, že tento trend sa nezmení ani v  najbližších rokoch a  teda náš kurz bude dlho aktuálny.

# 2  Ciele záverečnej práce

Cieľom práce je vytvoriť kurz so základnými informáciami v  oblasti strojového učenia. Týmto chceme priniesť jednoducho podaný zdroj na vlastné vzdelávanie, ktorý zvládne ktokoľvek so základnými znalosťami programovania, najlepšie v  jazyku Python. Chceme zároveň aj podnietiť záujem o  túto tému a  budúce štúdium.

Výslednou formou podania informácií bude kurz v  systéme Priscilla, ktorú opíšeme v  nasledujúcej kapitole. Avšak predmetom tejto práce nie je vkladanie a  ani úprava kódu pre potreby stránky.

Kurz bude obsahovať typickú štruktúru kurzu, čiže bude rozdelený do logických kategórií, tie budú obsahovať úvod, teoretické poznámky a  úlohy.

Teoretická stránka bude podaná jednoducho ale výstižne, bude obsahovať vizuálne pomôcky, ako zvýraznené časti, krátke poznámky v  odrážkach odrážok alebo obrázky.

Úlohy budú teoretické a  praktické. Teoretické úlohy budú spravené tak, aby sa odpoveď dala vyhľadať priamo v  kurze, ale zároveň poukazovali na dôležité časti alebo podnecovali otázky u  študenta na danú tému. Praktické otázky budú slúžiť na preopakovanie si kódu z  daných tém samostatne a  tým si študent bude môcť uvedomiť štruktúru programov ako aj čo je v  nich potrebné a  čo robia jednotlivé funkcie, parametre a  časti v  kóde.

Na záver každej kapitoly bude časť, ktorá odporučí ďalšie možnosti pokračovania v  štúdiu.

# 3 Návrh kurzu

Kapitola sa zameriava na samotný návrh kurzu. Analyzuje dostupné knižnice, popisuje nástroje pre tvorbu kurzu a  na záver rozpisuje základnú kostru kurzu.

## 3.1  Knižnice

V  programovacom jazyku Python je momentálne veľa knižníc zaoberajúcich sa problematikou umelej inteligencie. Knižnice uľahčujú prácu v  danej téme tým, že používateľ môže využívať vbudované funkcie.

Pri umelej inteligencii, v  našej práci, sme sa pozerali najmä na komplexnosť, jednoduchosť a  využitie.

Pre našu prácu sme vyberali spomedzi najviac využívaných knižníc s  cieľom, aby neboli priveľmi náročné na čítanie a  pochopenie a  zároveň sa ľahko učili. Nasledujúce knižnice sme zvažovali pre využitie a  zanalyzovali sme ich kladné a  záporné stránky.

### 3.1.1  Scikit-learn

Open-source knižnica Scikit-learn[[13]](#footnote-13), ktorá bola vybudovaná na pilieroch viacerých knižníc (NumPy, SciPy, matplotlib) a  to s  hlavným cieľom práce v  oblasti strojového učenia. Keďže využíva viaceré princípy z  iných knižníc, tak sa jej tvorca mohol zamerať na jednoduchosť a  prehľadnosť kódu, čo je hneď vidieť pri prvom pohľade, funkcie sú veľmi ľahko rozoznateľné a  ich používanie je jednoduché. Zameriava sa hlavne na prácu s  viacerými modelmi, čo bolo pre naše potreby dôležité. (Kramer, 2016)

### 3.1.2  TensorFlow

TensorFlow[[14]](#footnote-14) je knižnica vybudovaná spoločnosťou Google. Ide o  framework (aplikačný rámec), ktorý pracuje s  pokročilejšími neurónovými sieťami a  podporuje tvorbu AI aplikácií. (Shukla, Fricklas, 2018) Vo všeobecnosti ide skôr o  pokročilú knižnicu a  teda sa nehodila pre naše účely.

### 3.1.3  XGBoost

XGBoost[[15]](#footnote-15) je knižnica, ktorá pracuje primárne s  regresnými modelmi, napríklad často je používaná v  spojení so stromovými štruktúrami. Vyznačuje sa vlastnou technológiou boosting. Je to technológia, ktorá prináša vo veľa prípadoch jedinečné riešenia pri analýze stromových príkladov, ktoré majú hĺbku väčšiu ako 1 . (Chen a  kol., 2015)

Aj napriek popularite tejto knižnice sme ju pre našu prácu nevyužili a  to z  dôvodu, že nebola dostatočne komplexná a  nepokrývala témy, ktoré boli pre nás potrebné.

### 3.1.4  Keras Python

Keras[[16]](#footnote-16) je open-source knižnica (jej zdrojový kód je voľne prístupný) určená najmä na tvorbu neurónových sietí. Je veľmi prístupná pre začiatočníkov, umožňuje rýchle a  jednoduché vytváranie prototypov a  modelov. Taktiež podporuje prácu s  obrázkami a  textom. (Mosconi, 2019)

Pri prvotnej fáze zvažovania výberu témy kurzu bola našou voľbou, avšak neskôr sme smerovali kurz na strojové učenie a  stala sa menej vhodnou.

### 3.1.5  NumPy Python

NumPy[[17]](#footnote-17) alebo Numerical Python je knižnica v  prvom rade zameraná na výpočty. Oproti ostatným knižniciam vyniká v  numerických oblastiach, napríklad lineárna algebra, práca v  priestore n -dimenzií, náhodné čísla a  iné.(AI Publishing, 2021)

Je často krát považovaná za vstupnou bránou do strojového učenia, z  dôvodu využívania viac dimenzionálnych polí a  teda je vhodná na základné vedecké výpočty v  strojovom učení.

My sme sa rozhodli zahrnúť ju v  našom kurze. Aj keď jej potenciál sme nevyužili naplno, ale dali sme možnosť študentom poznať rôzne knižnice.

### 3.1.6  Theano Python a  PyTorch Python

Obe knižnice zrýchľujú proces počítania vďaka prístupu cez GPU a  taktiež sú vybudované z  knižnice NumPy. Theano[[18]](#footnote-18) je knižnica zameraná na strojové učenie a  tvorbu neurónových sietí. PyTorch[[19]](#footnote-19) sa zaoberá hlavne témou deep learning (hĺbkové učenie) a  dobre pracuje s  grafmi.(Sachdev, 2019)

Tieto knižnice neboli pre naše potreby vhodné, keďže išlo o  menej známe knižnice s  menším využitím vo sfére vyučovania strojového učenia.

### 3.1.7  Python Pandas

Knižnica Pandas[[20]](#footnote-20) je často krát spomínaná v  článkoch o  najviac používaných knižniciach avšak jej primárny smer pôsobenia je skôr spracovanie dát. Taktiež obsahuje veľké možnosti zobrazovania dát do grafov a  ich úpravu ako aj štatistické nástroje na analýzu. Veľkým plusom pri práci s  Pandas je štruktúra DataFrame[[21]](#footnote-21). Ide o  dvojdimenzionálnu tabuľku, ktorá sa pripodobňuje napríklad Excel tabuľkám (Obrázok 2 ). Obsahuje názvy stĺpcov a  riadkov a  môže obsahovať dáta rôznych typov. (McKinney, 2011)

Text

Description automatically generated

Obrázok 2 Ukážka štruktúry DataFrame[[22]](#footnote-22)

Pandas sme vybrali kvôli DataFrame, ktorý predstavoval zaujímavý prostriedok na prácu s  datasetmi (matica s  dátami). Taktiež táto knižnica je doteraz často využívaná v  strojovom učení.

### 3.1.8  Seaborn Python

Knižnica Seaborn[[23]](#footnote-23) je vytvorená nad knižnicou matlotlib. Matplotlib[[24]](#footnote-24) sa zameriava na vytváranie statických, animovaných a  interaktívnych vizualizácii dá, čiže popularita Seaborn stavia hlavne na tomto a  vytvára grafy s  témami, umožňuje úpravy syntax, aby bola jednoduchšia, vylepšuje rýchlosť a  stabilitu procesov v  pozadí.(Waskom, 2021) Seaborn sme si zvolili ako dodatočný prostriedok na vizualizáciu grafov.

### 3.1.9  Zhrnutie

Pre prácu sme si vybrali knižnicu Scikit-learn. Veľa článkov a  rozborov knižníc v  Pythone v  tejto oblasti dali Scikit knižnicu na druhé alebo nižšie miesto v  ich rebríčkoch hodnotenia. Zvyčajne bola odporúčaná pre začiatočníkov a  teda bola pre naše účely najvhodnejšia. Hlavné prednosti sme videli v  jej prístupnosti, prehľadnosti, flexibilnosti a  obsahu, ktorý pokrýva. Obsahuje veľké množstvo vbudovaných algoritmov, čiže sme sa nemuseli zdržiavať ich programovaním.

Dodatočne sme zvolili knižnicu Python Pandas, primárne kvôli jej práci s  dátami cez štruktúru DataFrame, ktorá je veľmi kompatibilná s  formou datasetov.

Nakoniec sme vybrali aj knižnicu Seaborn, ktorú sme využili na grafické vizualizácie kvôli jej vzhľadu a  funkčnosti. Pre rozšírenie kurzu o  túto knižnicu sme sa rozhodli z  dôvodu spestrenia vizualizácie a  teda hlbšieho pochopenia ako fungujú modely a  ako sa menia dáta.

## 3.2  Nástroje

Táto časť popisuje základné nástroje, ktoré sme využili pri tvorbe kurzu.

### 3.2.1  Anaconda

Anaconda[[25]](#footnote-25) je voľne dostupná open-source distribúcia pre programovacie jazyky Python a  R . Využíva sa na vedecké výpočty ako napríklad spracovávanie veľkého množstva dát, analýzu predikcií, ale aj strojové učenie. Veľkým plusom je, že obsahuje viac než 100 už vopred nainštalovaných balíkov, pripravených na prácu pre používateľa. Z  tohto dôvodu sme si aj vybrali prácu s  Anacondou.(Yan, Yan, 2018) Obsahuje taktiež už nainštalovaný a  nastavený Jupyter Lab. Pomocou tohto nástroja sme vedeli jednoducho nainštalovať všetko potrebné pre vývoj v  rámci kurzu. Čo sme videli ako výhodu aj pre študentov, pretože sa nemusia v  budúcnosti zaoberať zložitými inštalačnými postupmi jednotlivých balíkov a  knižníc.

### 3.2.2  Jupyter Lab

Jupyter Lab je webové používateľské rozhranie, ktoré sa nachádza v  Project Jupyter[[26]](#footnote-26), čo je projekt na vývoj voľno dostupného softvéru, ktorý podporuje prácu vo veľa programovacích jazykoch.

Umožňuje pracovať s  dokumentmi a  aktivitami ako Jupyter notebooks, textovými editormi, terminálmi a  s  vlastnými komponentami vo flexibilnom, integrovanom a  rozšírenom prístupe.(Granger, Pérez, 2021) Názov Jupyter vznikol spojeným troch programovacích jazykov Julia, Python a  R , celkovo podporuje viac než 40 jazykov.

Medzi jeho najväčšie výhody patrí napríklad, že sa dá spúšťať kód po blokoch a  tieto bloky môžu byť nezávislé na sebe navzájom, podporuje vizualizáciu dát a  v  neposlednom rade využíva Markdown jazyk, ktorý sa dá vkladať medzi bunky s  kódom a  tam písať poznámky.

V  našej práci sme vytvárali kurz v  Jupyter notebookoch a  využili sme funkcionalitu, ktorú sme popísali vyššie. Vo finálnej podobe sme vytvorili sme komplexné súbory, ktoré obsahovali kompletné kapitoly kurzu. Vďaka vlastnostiam notebookov sme zjednodušili budúcu prácu pri vkladaní kurzu do systému Priscilla. Keďže sme nemuseli využívať iné textové editory pre zadania, teóriu alebo poznámky, tak kurz vyzerá aj bez vloženia na cieľovú stránku prehľadne a  taktiež je možné ho vypracovať.

## 3.3  Priscilla

Stránka Priscilla je tvorená Univerzitou Konštantína Filozofa v  Nitre. Vznikla s  pomocou dotácií od Európskej Komisie pod programom ERAZMUS+. Na stránku sa môže registrovať ktokoľvek a  teda je prístupná pre verejnosť.

K  dnešnému dňu je stále vo vývoji, pridávajú sa tam nové kurzy. Momentálna ponuka je nasledovná:

Graphical user interface, application

Description automatically generated

Obrázok 3 Prehľad kurzov na stránke Priscilla – január 2022[[27]](#footnote-27)

Náš kurz sa zaradí do ponuky Umelej inteligencie, kde sa nachádza kurz na spracovanie prirodzeného jazyka, čo je ďalšia možnosť rozširovania si znalostí v  oblasti AI.

Na obrázku 3  vidíme, že sa na stránke nachádzajú aj rôzne programovacie jazyky, po otvorení nájdeme aj kategóriu Python. Takže v  ponuke sú aj rozsiahle kurzy na programovací jazyk Python- Python-začiatočník I  a  II, Python-pokročilý a  Python OOP, môžeme teda predpokladať, že k  nášmu kurzu môže študent absolvovať tieto kurzy a  tým sa naučiť potrebné znalosti na programovanie a  teda mať dostatočné úroveň programovanie na dokončenie nášho kurzu.

## 3.4  Kurz

Prvotná myšlienka pre náš kurz sa zakladala na tom, že bude približovať najmä ako funguje strojové učenie, avšak v  úvode sme plánovali predstavenie umelej inteligencie ako celku aby sme študentovi ukázali prepojenie medzi týmito pojmami.

Od začiatku tvorby sme predpokladali, že študenti začnú kurz s  dostatočnými znalosťami programovacieho jazyka Python a  teda nebude potrebné vysvetľovať programovaciu stránku.

Vízia bola, že kurz bude vedený pre začiatočníkov a  bude objasňovať základné modely a  princípy strojového učenia, takže ani z  hľadiska programovania nebude náročný. Taktiež, že kurz bude zostavený zo sekcií, každá bude mať úvod do teórie a  praktické príklady. Študenti si budú môcť vyskúšať samotné naprogramovanie úloh rôznych modelov strojového učenia. Predpokladali sme, že pre kurz bude prístupná funkcionalita stránky Priscilla a  teda naprogramované úlohy budú kontrolované automaticky testami a  následne dajú hodnotenie vizuálne aj slovné. Pri sekciách sme plánovali napísať úvodné texty ako aj zadania a  sprievodné texty k  úlohám.

Pri tvorbe sme chceli zachovať myšlienku inšpirácie študenta, o  jeho zamyslenie sa nad problémami a  privedenie ho k  novým spôsobom pozerania sa na svet a  každodennému využitiu AI. Tento cieľ sme podporili pridanými zdrojmi za sekciami, keďže veríme, že pomôžu naviesť budúcich študentov k  ďalšiemu štúdiu.

# 4 Postup vytvárania kurzu

Táto kapitola sa zameriava na proces tvorby kurzu. Popisuje aké knižnice sme použili a  aké metódy sme zvolili pri jednotlivých kapitolách kurzu.

## 4.1  Využitie knižníc

Pri analýze riešenia, keď sme plánovali, ktoré knižnice budú vhodné, sme vybrali ako hlavnú knižnicu Scikit-learn. Táto knižnica nám aj po vypracovaní kurzu prišla ako vhodná voľba. Avšak v  niektorých prípadoch sme využili iné knižnice a  tie prípady sme popísali neskôr v  tejto kapitole.

V  prvom rade treba poukázať na to, že knižnica Scikit-learn bola plne postačujúca na náš kurz. Obsahovala všetky potrebné modely aj funkcie, čiže by sa dal celý kurz vypracovať iba v  nej. Najväčšiu výhodu sme videli v  tom, že bola jednoduchá na pochopenie. Čo bolo a  aj je kľúčové pri kurzoch cielených na začiatočníkov. Je to z  dôvodu, že jej syntax je jednoduchá na popísanie a  funkcie sú ľahké na použitie.

A picture containing text

Description automatically generated

Obrázok 4 Ukážka použitia knižnice Scikit-learn[[28]](#footnote-28)

Obrázok 4  je z  nášho kurzu, kde sme popisovali metódu imputácie v  téme predspracovania dát. Z  obrázku je vidieť, že Scikit-learn je jednoduché prečítať aj bez bližšej znalosti teórie a  všetky metódy aj parametre sú jednoznačne čitateľné.

V  niektorých momentoch sme si uvedomili, že dáta vizualizované pomocou Scikit-learn sú strohé a  nie sú pekné na oko. Tak sme sa rozhodli pre vizualizáciu pomocou knižnice Seaborn. Grafy z  tejto knižnice boli vo finále farebné a  obsahovali aj popisy (obrázok 5 ). Celkovo oživili kurz a  to bolo cieľom pri ich využití.

Chart, scatter chart

Description automatically generated

Obrázok 5 Ukážka použitia knižnice Seaborn[[29]](#footnote-29)

Pri použití knižnice iba pre krajšiu vizualizáciu dát sme si museli uvedomiť, či chceme aby bola táto knižnica aj preberaná v  kurze. Čiže, či sme chceli venovať kapitolu alebo podkapitolu tejto knižnici a  popísať ju podrobnejšie. Rozhodli sme sa, že tak nespravíme a  budeme považovali sme ju iba za sprievodnú knižnicu. Čo znamenalo, že sme samotný kód nepopisovali a  nevyžadovali sme ho v  úlohách a  ani sme nevyžadovali aby si kód pamätali študenti a  aby ho vedeli použiť. Dokonca sme brali do úvahy aj možnosť, že by v  budúcnosti, pri práci na stránke Priscilla, boli tieto kódy nahradené iba obrázkami. Pre túto možnosť sme sa avšak pri tvorbe nerozhodli, aby sme dali možnosť študentovi pozrieť sa aj na funkcie iných knižníc a  ako sa dajú využiť v  praxi.

Okrem knižníc Scikit-learn a  Seaborn sme využili ešte Python Pandas. Táto knižnica by dokázala plne nahradiť Scikit-learn a  dal by sa na nej vybudovať kompletný kurz, ale my sme sa pre ňu nerozhodli, lebo je vo všeobecnosti viacej komplexná a  obsiahlejšia. Čo sme považovali, že by bolo na úkor množstva kódu. Vo veľa prípadoch sme využili Pandas ako dodatočnú knižnicu na rozšírenú funkcionalitu a  taktiež na prácu s  datasetom, ktorý sme importovali zo stránky (obrázok 6 ).

Graphical user interface, text, application

Description automatically generated

Obrázok 6 Ukážka použitia knižnice Pandas[[30]](#footnote-30)

Doteraz sme popisovali hlavné knižnice. V  kurze je použitá v  kapitole o  predspracovaní dát aj knižnica NumPy. Nepovažovali sme za potrebné ju spomínať ako hlavnú knižnicu, keďže sme ju využili iba ako spôsob na zadefinovanie matice. Čo znamená, že sme ju nepoužili priamo na demonštráciu strojového učenia, ale skôr iba ako nástroj na prípravu dát (obrázok 7 ).

Graphical user interface, application, Word

Description automatically generated

Obrázok 7 Ukážka použitia knižnice NumPy[[31]](#footnote-31)

## 4.2  Proces výberu tém pre kurz

Táto časť v  tvorbe kurzu bola kľúčová. Bolo potrebné určiť hranicu medzi jednoduchým, ľahko pochopiteľným kurzom, ktorý by obsahoval základ pre pochopenie strojového učenia a  príliš komplexným a  do hĺbky sa zaoberajúcim kurzom. V  konečnom dôsledku sme pri tomto bode strávili veľa času analýzou a  pri vytváraní kurzu sme viac krát menili jeho obsah.

Pôvodne sme plánovali obsiahnuť iba úvod, predstavenie dvoch modelov a  prácu s  nimi a  následne kapitolu o  testovacích a  trénovacích dátach, avšak už pri tvorbe kapitol o  modeloch, sme narazili na nevysvetlené poznatky o  datasetoch. Neskôr sme doplnili kurz o  ďalšie témy a  finálna verzia obsahovala tieto kapitoly:

### 4.2.1  Úvod do kurzu

Úvod sme zvolili krátky, keďže ide o  kurz a  nie o  knižnú publikáciu. Nepovažovali sme potrebné písať ho rozsiahly. Počítali sme s  tým, že študenti majú vlastnú motiváciu siahnuť po kurzu a  teda sme popísali iba o  čom je náš kurz a  aký sme mali vlastný cieľ pri jeho tvorbe.

Na koniec sme napísali krátky obsah, aby študent vedel zhodnotiť, či je pre neho tento kurz vhodný.

### 4.2.2  Knižnice

Počas práce na kurze sme pracovali najviac s  knižnicami Scikit-learn, Pandas a  NumPy. Tieto knižnice sme viacej popísali v  tejto kapitole s  cieľom predstaviť ich na začiatok študentovi.

Preto sa táto kapitola nachádza aj na začiatku, aby študent vedel, čo môže očakávať a  pozrieť sa trochu bližšie na teoretickú funkcionalitu knižníc pred tým, ako ich uvidí a  využije v  praxi.

Zvažovali sme, či zahrnieme aj knižnicu Seaborn, ale rozhodli sme sa pre jej vynechanie z  dôvodu, že túto knižnicu sme nevyužili vo veľa prípadoch a  v  tých pár nebola súčasťou úlohy. To znamená, že sme nevyžadovali, aby ju študent pochopil do hĺbky a  vedel s  ňou pracovať. Slúžila nám iba na vizuálne zobrazenie dát.

Taktiež sme brali do úvahy možnosť, že sa po vložení kurzu na stránku Priscilla kód úplne vynechá. Čiže môže byť viditeľný iba graf.

### 4.2.3  Iris dataset

Po vypracovaní kapitol o  modeloch, predspracovaní dát a  o  porovnávaní modelov a  trénovacie a  testovacie dáta, sme si uvedomili to, že využívame najviac dataset Iris. Teda sme videli potrebu popísať ho bližšie.

Bolo našim cieľom pracovať s  Iris. Chceli sme sa sústrediť na vysvetľovanie procesov a  princípov a  nechceli sme často písať o  čom je dataset a  ako sa s  ním pracuje. Tým by sme kapitoly značne predĺžili a  predpokladali sme, že by stali menej prehľadné.

Takže sme sa rozhodli, že vypracujeme samostatnú kapitolu o  datasete Iris. Cielili sme na to, aby bolo na začiatku vysvetlené, čo obsahuje tento dataset.

Tento dataset sme zvolili z  dôvodu, že je jednoduchý na pochopenie a  obsahuje dáta, na ktorých sa dajú ukázať mnohé metódy.

V  samotnej kapitole sme na úvod predstavili pojem dataset a  následne sme popísali štruktúru datasetu Iris. Na tejto štruktúre sme zaviedli názvy, ktoré sa využívajú v  strojovom učení a  my sme ich používali aj neskôr v  kurze. Z  tohto dôvodu sme potrebovali upriamiť pozornosť na ich zapamätanie a  teda sme pripravili kontrolnú otázku na overenie týchto pojmov.

Síce sa táto kapitola volá dataset Iris, ale predstavujeme tu prvý krát aj prácu s  datasetom ako takým. Popísali sme presný postup importu ako aj základných metód, ktoré predstavili dáta.

Keďže už počas úvodu si mohol študent položiť otázku, na čo má dataset Iris uložené konkrétne názvy druhov kvetín ako oddelené pole, tak sme túto otázku priamo vyslovili a  odpovedali na ňu.

K  tomu sme zostavili kontrolnú otázku, aby študent vedel, či pochopil túto problematiku a  my sme ju dobre vysvetlili. Celkovú prácu s  datasetom si následne mohol otestovať v  programovacej úlohe.

Úplne na koniec sme sa zamerali na samostatnú vizualizáciu dát datasetu. Keďže sme v  celej kapitole využívali iba knižnicu Scikit-learn, tak sme ukázali viac prehľadné a  krajšie zostavené grafy. Týmto sme predstavili aj knižnice Pandas a  NumPy v  praxi.

### 4.2.4  Predspracovanie dát

Počas tvorby kapitol o  modeloch sme si uvedomili potrebnosť predspracovania dát. Najskôr sme chceli zahrnúť celý proces tvorby vlastného datasetu, určovania toho, či sú dáta vhodné a  na čo si treba dávať pozor, ale nakoniec sme sa rozhodli to veľmi zjednodušiť. V  tejto kapitole sme poukázali na to, že dataset je len matica a  nechali sme na študentovi, nech sa zamyslí, či akákoľvek a  či bude mať výpovednú hodnotu, ak by obsahovala nevhodné dáta. Pre lepšie pochopenie tvorby datasetu sme do zdrojov pridali stránku na jeho tvorbu.

V  úvode kapitoly sme zodpovedali na nami položenú otázku, na čo nám je predspracovanie dát a  pre ľahšie porozumenie sme pridali jednoduché príklady.

Po vytvorení veľmi jednoduchého datasetu, kde sme nepozerali na logickú stránku, sme predstavili základné techniky. Týmito technikami sme ukázali ako opraviť dáta, ktoré majú chybné hodnoty, tie sme predstavili v  úvode.

Kategórie sme zvolili podľa techník, ktoré sa momentálne využívajú a  opravujú najčastejšie problémy pri hodnotách v  datasetoch.

Najskôr sme predstavili imputáciu, ktorá sa využíva keď nemáme nadefinované hodnoty v  datasete. Pri všetkých ukážkach sme sa snažili zvoliť veľmi jednoduché matice-datasety tak, aby priamo poukazovali na daný problém, ktorým sme sa zaoberali. Čiže sme nekombinovali techniky navzájom, ale nechali sme ich oddelené. Cieľom bolo ľahšie vidieť rozdiely a  pochopiť význam techník. Taktiež sme sa snažili predstaviť ako sa delia dané techniky napríklad pri imputácii, že sa delí podľa spôsobu fungovania na priemernú hodnotu, medián a  najčastejšiu hodnotu.

Ako druhú sme zvolili techniku mapovania a  to z  dôvodu, že je pomerne jednoduchá a  dá sa jasne vysvetliť.

Najzložitejšiu techniku a  to techniku škálovania sme si nechali na koniec. Keďže je táto technika náročnejšia, tak sme k  nej pripravili aj teoretickú otázku, kde sme sa upriamili na to, či študent chápe rozdiel medzi dvoma škálovacími technikami.

Na koniec kapitoly sme pripravili programovaciu úlohu, aby si študent odskúšal aj praktickú stránku škálovania.

### 4.2.5  Učenie s  učiteľom

Keďže sme primárne chceli objasniť modely KNN a  lineárnej regresie, tak sme museli pridať kapitolu, ktorá by teoreticky objasňovala aké sú rozdiely medzi regresiou a  klasifikáciou a  teda aj medzi týmito modelmi.

Kapitolu sme nechceli obsiahlu a  po jej dopísaní sme zvažovali, že ju pridáme k  inej, ale bolo potrebné, aby bolo zrejmé, že je to celok a  teda si študent môže kedykoľvek s  ľahkosťou nájsť túto kapitolu a  pozrieť si ju.

Na úvod sme popísali teoretické základy rozdielov medzi klasifikáciou a  regresiou, ktoré obsahovali aj pojmy kvantitatívne a  kvalitatívne hodnoty, tie sme objasnili po úvode.

Taktiež sme upozornili na to, že neplatí, že ide o  kvalifikáciu ak máme kvantitatívne hodnoty a  podobne pri regresii a  kvalitatívnych hodnotách. Tento fakt sme považovali potrebný zdôrazniť, aby si študent tak nemyslel.

Keďže kapitola je krátka, tak sme pridali len jednu overovaciu teoretickú otázku na pochopenie rozdielu medzi klasifikáciou a  regresiou. Vybrali sme veľmi základný a  často spomínaný príklad s  emailovou poštou a  jej delením na spam a  not-spam.

### 4.2.6  KNN

Na začiatok kapitoly bolo potrebné popísať prečo sme si vybrali práve tento model. Keďže sme chceli mať príklad modelu z  klasifikácie aj regresie a  zároveň vybrať modely, ktoré sú rozšírené a  dajú sa popísať a  odskúšať jednoducho, tak sme pre klasifikáciu vybrali model K -najbližších susedov (KNN).

Model funguje na princípe vytvárania oblastí so spoločnými vlastnosťami. Pri popisovaní princípu fungovania sme použili aj obrázky, ktoré sme vytvorili v  programe Microsoft Paint. Týmto sme sa snažili najmä priblížiť súvislosti a  lepšie predostrieť o  čom píšeme v  texte a  teda o  samostatnom princípe modelu.

Zároveň sme na začiatku aj použili vykresľovanie grafu cez knižnicu Seaborn, ktorú sme spomínali v  jednej z  predošlých kapitol.

Po úvode, kde študenti získal predstavu, ako funguje KNN, sme prešli na popis ako si sami vedia naprogramovať KNN.

Potrebovali sme využiť už predtým spomenuté premenné *X* a  *y*, tak sme smerovali teoretickú otázku na toto overenie si znalostí. Čiže sme študentovi aj nepriamo ponúkli test, či si pamätá základné avšak dôležité znalosti, ktoré sme popísali už skôr v  kurze.

Po vysvetlení teoretickej stránky ako naprogramovať vlastný KNN model na datasete Iris, sme pripravili programovaciu úlohu. Tu si študent vedel overiť, či si pamätá programovanie KNN a  či rozumie aj čo robia jednotlivé riadky kódu. Pri tejto úlohe sme našli dve hlavné riešenia, ktoré sme zahrnuli do sekcie riešenia.

### 4.2.7  Lineárna regresia

Okrem klasifikačných problémov sa v  strojovom učení riešia aj regresné. Jeden z  najzakladanejších regresných modelov je lineárna regresia a  pre tento dôvod sme ju zvolili ako zástupcu a  popísali sme ju v  tejto kapitole.

V  úvode sme vysvetlili, čo je to lineárna regresia a  využili na to dovtedy nespomenutý dataset o  reklamách. Ten sme museli na úvod popísať, objasniť ako funguje a  aké dáta obsahuje. Pri tom sme pripravili aj kontrolnú otázku z  teórie metód. Taktiež sme zahrnuli aj trénovacie a  testovacie dáta, pri ktorých sme potrebovali spomenúť, že sa im venujeme v  nasledujúcej kapitole. Takýmto prepojeniam sme sa chceli čo najviac vyhnúť, aby boli kapitoly samostatnými celkami. Niekedy to avšak nešlo a  namiesto opakovania sa, sme zvolili prístup, že spomenieme kapitolu, kde sa danej problematike venujeme.

Na konci, po predstavení ako funguje lineárna regresia, sme potrebovali popísať, k  akým výsledkom sme sa dopracovali. Keďže tento dataset nemá úplne zrejmé výsledky. Pre lepšie pochopenie sme vykreslili grafy za pomoci knižnice Seaborn.

### 4.2.8  Porovnávanie modelov a  trénovacie a  testovacie dáta

Pre túto kapitolu sme sa rozhodli, aby sme ukázali na jednom mieste ako sa dajú meniť činnosti modelov a  teda ako vieme ovplyvňovať ich výsledky.

Pôvodne sme chceli zahrnúť viac modelov ako KNN a  lineárnu regresiu, ale po uvážení toho, že by sme museli každý odznova vysvetľovať a  tým sa náš kurz o  základoch zväčšil na kurz komplexný, sme sa rozhodli demonštrovať zmeny na už dovtedy predstavených modeloch.

V  úvode sme opísali metódu train\_test\_split, ktorú sme využili na modeloch. Pri demonštrácii funkcií sme poukázali na zmeny parametrov modelov a  taktiež aj na to, že je potrebné dávať pozor na výpočtovú zložitosť a  krivku účinnosti.

Pre odskúšanie sme navrhli dve programovacie úlohy a  na úplný záver sme zhrnuli prečo boli opísané delenia na trénovacie a  testovacie dáta užitočné a  čomu predchádzame pri ich využívaní.

### 4.2.9  Záver kurzu

V  závere sme chceli najmä vyzdvihnúť to, že sme mali za cieľ prejaviť záujem u  študenta o  strojové učenie a  podnietiť jeho ďalšie skúmanie. Pre podporenie tohto cieľa sme poukázali na dodatočné zdroje, z  ktorých môže čerpať v  budúcnosti.

# 5 Výsledky riešenia a  ich zhodnotenie

Táto kapitola popisuje jednotlivé výsledky, ktoré sme nadobudli počas tvorby kurzu a  zároveň analyzuje hodnotenia dotazníkov od študentov.

## 5.1  Práca v  Jupyter notebookoch

Voľba práce v  Jupyter notebookoch bola v  konečnom dôsledku správna. Jednoznačne sme videli všetky klady, ktoré táto štruktúra ponúka a  ktoré sme popísali v  kapitole 3 .2 .2 .

Vo finálnej podobe sme mali jednoduché a  elegantné riešenie pre náš kurz, kde sme kombinovali časti pre kód a  text (obrázok 8 ) a  postupovali sme podľa plánovaného obsahu kurzu. Po textových častiach, ktoré ozrejmovali teoretickú časť sme vkladali kód, ktorý dopĺňal prakticky napísaný text. Vedeli sme takto vypracovať celý.

Veľmi sme ocenili a  aj využili možnosti Markdown jazyku, v  ktorom sme písali textové časti. Markdown obsahuje kódovanie pre zošikmenie alebo zhrubnutie textu, ako aj rôzne veľkosti pre nadpisy (obrázok 8 ). Využívaním týchto vizuálnych pomôcok sme sa snažili zjednodušiť orientáciu v  kurze a  zároveň uľahčiť jeho pochopenie.

Graphical user interface, text, application, email

Description automatically generated

Obrázok 8 Ukážka kombinovania kódu a  textu[[32]](#footnote-32)

## 5.2  Dotazník a  kurz v  praxi

Vďaka súhlasu Mgr. Ľubomíra Benka Ph.D . sme mali možnosť odskúšať kurz na predmete Programovacie jazyky pre umelú inteligenciu. Využili sme na to priestor jedného cvičenia.

Na cvičení vyplnilo sedem študentov náš kurz, ktorý bol pre nich pripravený pomocou nástroja Colab[[33]](#footnote-33) od spoločnosti Google. Taktiež vyplnili sprievodný dotazník, za pomoci ktorého sme dostali spätnú väzbu na vyhodnotenie.

Dotazník bol vytvorený pomocou nástroja Forms taktiež od spoločnosti Google. Túto formu sme si vybrali z  dôvodu jednoduchosti nástroja, ktorý nám taktiež ponúka štatistické grafy výsledkov. Zároveň je to najrozšírenejšia forma dotazníkov a  mohli sme predpokladať, že všetci študenti sa s  ňou už stretli a  prostredie dotazníka im teda nebolo cudzie.

Cieľom dotazníka bolo, aby sme zistili možné chyby a  vylepšenia do budúcna. Avšak testovaná vzorka študentov bola malá a  teda sa nedá považovať za štatisticky významnú. Výsledky sme ale zanalyzovali a  popísali sme výsledky nižšie.

### 5.2.1  Vzorka študentov

V  úvode dotazníka sme sa pýtali na základné informácie. V  nich sme pozorovali na akej úrovni vzdelania v  oblasti programovania a  umelej inteligencie študenti sú. Nepredpokladali sme expertné znalosti, keďže sme podávali dotazník na cvičení o  úvode do umelej inteligencie.

Naše predpoklady sa potvrdili. Nemali sme ani jedného študenta, ktorý by vedel o  strojovom učení do veľkej hĺbky. Väčšina sa zaradila dokonca do kategórie, že by vedeli len niečo povedať o  tejto téme a  teda si neveria ani na základné poznatky.

Výnimočne dobrá bola spätná väzba na otázku, či sa zaujímajú o  strojové učenie. Mali sme sto percentnú odpoveď áno, z  čoho sme vyvodili chuť a  záujem o  vyplnenie kurzu a  teda sme mohli očakávať konštruktívnu kritiku.

### 5.2.2  Úvodná kapitola a  knižnice

Po predošlej veľmi kladnej odpovedi o  záujem sme dostali aj kladné opísanie tém kurzu v  úvode.

Chart, pie chart

Description automatically generated

Obrázok 9 Záujem študentov o  témy v  kurze[[34]](#footnote-34)

Až na jedného študenta vyzerali témy zaujímavo a  teda boli lákavé na štúdium (obrázok 9 ).

Chart, pie chart

Description automatically generated

Obrázok 10 Znalosť knižníc v  kurze[[35]](#footnote-35)

Chart, pie chart

Description automatically generated

Obrázok 11 Miera zrozumiteľnosti popísania knižníc[[36]](#footnote-36)

Z  obrázku 10 usudzujeme, že študenti, ktorí odpovedali, že poznali aspoň jednu opísanú knižnicu, tak sa im zdala aj táto kapitola zrozumiteľná, ako je vidieť na obrázku 11.

Do budúcna by sme navrhovali pridať viac ukážok funkcionality jednotlivých knižníc alebo programovacie úlohy. Taktiež by sa dal kurz prerobiť tak, aby využíval iba jednu knižnicu. Naše rozhodnutie pre viacero knižníc bolo s  myšlienkou, že naučíme študentov väčšieho množstva funkcionality a  poukážeme na to, že sa nemusí všetko vykonávať iba v  jednej knižnici. Ukázalo sa, že to môže byť skôr mätúce ako prínosné a  mali by sme skôr dodržiavať ciele kurzu a  ponechať knižnice na neskoršie kurzy, ktoré by sa im mohli aj venovať podrobnejšie.

### 5.2.3  Zrozumiteľnosť

Táto otázka smerovala na to, či boli jednotlivé kapitoly zrozumiteľné a  pochopiteľné.

Chart, bar chart

Description automatically generated

Obrázok 12 Miera zrozumiteľnosti naprieč témami[[37]](#footnote-37)

Z  obrázku 12 môžeme vidieť, že kurz akoby nadobúdal vyššiu obťažnosť kapitolu po kapitole. Naše úvahy boli, že to bolo spôsobené veľa informáciami, ktoré sa často nabaľovali počas priechodom kurzom. Riešením by mohlo byť to, že by študenti mali prestávky v  kurze. Napríklad, že by bolo jeho vypĺňanie na týždennej báze. Taktiež by mohlo byť riešením pridať na koniec každej kapitoly krátku rekapituláciu. Tá by mohla byť aj formou testu, čiže by študent mal pohromade všetky dôležité informácie a  overil by si, ako si ich pamätá.

### 5.2.4  Náročnosť

Táto otázka bola smerovaná konkrétne na teoretické a  praktické úlohy.

Chart, bar chart

Description automatically generated

Obrázok 13 Miera náročnosti naprieč témami[[38]](#footnote-38)

S  výsledkom sme boli veľmi spokojní, keďže sme sa snažili o  vyváženosť a  podľa odpovedí sa nám to podarilo (obrázok 13).

### 5.2.5  Množstvo úloh

Táto otázka bola smerovaná na potrebu precvičenia si teoretickej a  praktickej časti kurzu.

Chart, bar chart

Description automatically generated

Obrázok 14 Porovnanie počtu úloh v  témach[[39]](#footnote-39)

Podľa obrázku 14 vidíme priestor na zlepšenie v  tomto smere. Podľa viacerých študentov bolo vo veľa prípadoch úloh málo a  dokonca sme pozorovali, že čím neskoršia kapitola, tým viac študentom prišlo množstvo úloh nízke.

Z  obrázkov 12 a  14 vidíme prepojenie medzi zrozumiteľnosťou a  počtom úloh, je možné, že ak by sme dali viacero úloh, tak by sme mali aj vyššiu mieru pochopenia danej problematiky.

Náš cieľ pri úlohách bol, aby ich práve nebolo priveľa. Chceli sme skôr dať menej a  komplexných úloh, kde by sme vyžadovali zamyslenie sa nad teóriou alebo napísanie už komplexného kódu. Toto rozhodnutie bolo pri našej testovanej skupine chybné, ale musíme počítať s  tým, že v  budúcnosti bude kurz dostupný verejnosti a  môže teda nájsť väčšie množstvo ľudí s  hlbšími znalosťami v  oblasti informatiky. Tí by mohli mať názor na úlohy iný.

### 5.2.6  Množstvo teórie

Otázka smerovala na množstvo popisov a  textov, ktoré vysvetľujú teóriu. Zároveň aj to, či bola téma vysvetlená zrozumiteľne.

Chart, bar chart

Description automatically generated

Obrázok 15 Miera porozumenia teoretickej stránky naprieč témami[[40]](#footnote-40)

Obrázok 15 bol vytvorený cez automatické generovanie grafov na stránke Google Forms, posledný výsledok nemá popis, ale ide o  ôsmu kapitolu: Porovnávanie modelov a  train test.

Celkový výsledok bol dobrý, z  výsledkov bola teória pochopiteľná, až na drobné nedostatky. Tie by sa mohli odstrániť pri ďalšom prejdení kurzu alebo ak by nebol kurz riešený naraz. Taktiež riešením by mohla byť rekapitulácia alebo už vyššie spomenutý test na konci kapitoly.

### 5.2.7  Po vypracovaní kurzu

Táto časť bola veľmi dôležitá, smerovali sme otázky na to, či študenti považovali kurz za zaujímavý, či by si ho zopakovali a  hlavne, či sa niečo naučili.

Chart, pie chart

Description automatically generated

Obrázok 16 Rozloženie záujmu o  obsah kurzu[[41]](#footnote-41)

Chart, pie chart

Description automatically generated

Obrázok 17 Návratnosť ku kurzu[[42]](#footnote-42)

Nadpolovičná väčšina považovala kurz za zaujímavý (obrázok 16), taktiež rovnaké percento by sa aj vrátilo k  jeho zopakovaniu (obrázok 17). Považovali sme to za úspech, z  dôvodu, že kurz neodradil študentov.

Chart, pie chart

Description automatically generated

Obrázok 18 Prínos kurzu[[43]](#footnote-43)

Výborné výsledky sme mali v  otázke, či bol kurz prínosný pre študentov (obrázok 18). Usúdili sme, že až na jedného študenta si každý našiel niečo, čo sa naučil.

Ak by sa robila analýza na väčšej vzorke študentov, odporučili by sme zahrnúť otázku: „Konkrétne ktoré kapitoly boli najviac informačne prínosné“. Avšak pri väčšom počte študentov by sa pravdepodobne výsledky na túto otázku až tak nelíšili ako v  našom príapde a  teda by nemala veľký štatistický význam.

Chart, pie chart

Description automatically generated

Obrázok 19 Náročnosť kurzu[[44]](#footnote-44)

Vzhľadom na to, ako študenti odpovedali na náročnosť kurzu (obrázok 19), by sme do úvodu kurzu na stránke napísali, že ho odporúčame pre študentov vyšších ročníkov. Keďže väčšina študentov bola z  prvého ročníka bakalárskeho štúdia, tak by sme ho odporúčali možno do druhého alebo tretieho ročníka rovnakého stupňa štúdia. Za uváženie stojí aj magisterské štúdium, kde by mohol byť kurz na preopakovanie si strojového učenia pri predmetoch, ktoré idú viac do hĺbky v  tejto tematike.

Chart, pie chart

Description automatically generated

Obrázok 20 Odporúčanie kurzu[[45]](#footnote-45)

Aj napriek náročnosti by veľa študentov odporučilo náš kurz (obrázok 20). Zo spätnej väzby sme dostali dôvody, že je to dobrý začiatok, dokonca dobrý praktický úvod. Taktiež, že naučí základy a  je to zaujímavá forma na učenie. Dokonca, že všetko bolo zrozumiteľne vysvetlené. Kladné ohlasy mali aj dodatočné materiály na štúdium na konci kapitol, pri ktorých sme zvažovali, počas tvorby kurzu, či ich tam zahrnieme.

Z  negatívnejšej spätnej väzby sme dostali, že niekedy bolo ťažšie porozumieť témam a  hodilo by sa rozobrať témy viac do hĺbky. Toto zodpovedalo aj predošlým odpovediam pri otázkach a  teda sme to považovali za miesto na budúce zlepšenie.

### 5.2.8  Spätná väzba

Okrem už výčitiek spomenutých vyššie, sme dostali jednu na použitie datasetu o  kvetinách. V  budúcnosti by sa dal nahradiť dataset Iris aj iným, ale tento dataset sa nám zdal ako najvhodnejší na vysvetlenie a  nechceli sme opisovať v  každej kapitole nový dataset, aby to nespôsobovalo chaos. Chceli sme poukázať týmto prístupom na to, že dané metódy sa dajú praktikovať aj na jednom vhodne vybudovanom datasete. Avšak vedeli by sme priložiť aj iné datasety, iba na ukážku alebo spraviť praktické úlohy s  novými datasetmi, čo by vyriešilo aj dopyt po väčšom množstve úloh.

Taktiež veľa študentov poukázalo na to, že na väčšie pochopenie kurzu by potrebovali viac času. Testovanie kurzu prebiehalo na jednom cvičení, čiže sa dá povedať, že po úvodných informáciách, mali študenti na kurz približne hodinu.

Kurz nebol navrhovaný na takto krátku dobu. Môžeme teda z  tejto spätnej väzby odporučiť, aby študenti mali viac času na jeho vyplnenie. Hlavne z  dôvodu, aby mali priestor na hlbšie pochopenie tém.

### 5.2.9  Záver dotazníka

S  výsledkami z  dotazníkov a  vypracovania kurzu sme veľmi spokojní. Spätnú väzbu sme dostali konštruktívnu a  na základe nej sme vyššie navrhli riešenia do budúcnosti, avšak vzorka bola malá a  teda pred úpravami by sme navrhli otestovať kurz na širokej vzorke so študentmi s  viacerými znalosťami v  oblasti informatiky.

# Záver

Cieľom bakalárskej práce bolo vytvorenie kurzu pre výučbu umelej inteligencie v  programovacom jazyku Python. Počas tvorby práce sme si stanovili obmedzenia a  kľúčové body, ktoré sme sa pri implementácii snažili docieliť.

Kurz sme cielili na začiatočníkov v  umelej inteligencii so základnými znalosťami Pythonu. Upriamili sme sa na strojové učenie, kde sme popísali základné modely a  princípy, ako napríklad predspracovanie dát, učenie s  učiteľom. lineárna regresia, trénovacie a  testovacie dáta a  porovnávanie modelov. Snažili sme sa obsiahnuť témy, ktoré tvoria základ pre prácu v  strojovom učení a  hlavne priniesť kurz, ktorý by nebol náročný a  priniesol dostatok informácii študentom, aby pokračovali v  štúdiu strojového učenia.

Kurz bol vytvorený v  Jupyter notebookoch, kde sme využili ich štruktúru tak, aby sme vytvorili prehľadný a  vizuálne pekný kurz na okamžité použitie aj keď s  obmedzenou funkčnosťou. Úplnú funkčnosť sme nepotrebovali docieliť, keďže kurz bol plánovaný na implementáciu na stránku Priscilla, kde sa zaradí do programu výučby Umelej inteligencie a  stránka plne sfunkční obsah kurzu.

Na základe testovania kurzu na malej vzorke študentov sme navrhli rôzne prístupy a  odporúčania pri ďalšej úprave kurzu. Najmä sme vyzdvihli, že je potrebné kurz prechádzať postupne a  nie naraz, keďže jeho obsah je náročnejší a  jeho výstupnú hodnotu by sme obohatili o  rekapitulácie na konci kapitol. Taktiež by sme cielili kurz na študentov z  vyšších ročníkov.

Cieľ bakalárskej práce sme splnili, vytvorili sme funkčný kurz, ktorý je tematicky uzatvorený. Obsahuje praktické aj teoretické otázky a  pracuje s  viacerými knižnicami, kde sme najviac využívali Scikit-learn.

# Zoznam bibliografických odkazov

AI PUBLISHING. 2021. *Python NumPy for Beginners: NumPy Specialization for Data Science.* AI Publishing, 2021. 417 s . ISBN 978-1 -956591-09-5

Anderson, D ., 1989. *Artificial intelligence and intelligent systems: The implications*. Halsted Press, 1989. 178 s ., ISBN 0470214082

Anderson, J ., Rainie, L ., Luchsinger, A ., 2018. *Artificial intelligence and the future of humans.* V : *Pew Research Center* [online]. Pew Research Center, č . 10, s . 12. [cit. 2021-11-05]. Dostupné na internete <https://www.pewresearch.org/internet/2018/12/10/artificial-intelligence-and-the-future-of-humans/>

Ceder, N ., 2018. *The quick Python book*. Simon and Schuster, 2018. 478 s ., ISBN 9781638353430

Dick, S ., 2019. Artificial intelligence. In : *Harvard Data Science Review*, vol. 1 .1 . DOI 10.1162/99608f92.92fe150c

Etzioni, A ., Etzioni, O ., 2016. Keeping AI legal. In : *Vand. J . Ent. & Tech. L .*, č . 19, s .133. DOI 10.2139/ssrn.2726612

GRANGER, B., PÉREZ, F., 2021. *Jupyter: Thinking and storytelling with code and data.* V: *Published in Computing in Science & Engineering.* [online]. Vol. 23. i. 2. P. 7-14. DOI 10.22541/au.161298309.98344404/v3. Dostupné na internete <https://www.authorea.com/doi/full/10.22541/au.161298309.98344404>

Hassabis, D ., 2017. *Artificial intelligence: chess match of the century.* In : *Nature*, vol. 544, no. 7651, pp.413-414 [cit. 2022-01-10]. DOI https://doi.org/10.1038/544413a. Dostupné na internete <https://www.nature.com/articles/544413a>

Hong, J .W ., Cruz, I ., Williams, D ., 2021. *AI, you can drive my car: How we evaluate human drivers vs. self-driving cars.* V : *Computers in Human Behavior* [online]. vol. 125, p .106944. [cit. 2022-01-05]. DOI 10.1016/j .chb.2021.106944. Dostupné na internete <http://dmitriwilliams.com/wp-content/uploads/2021/08/Hong-2021-Ai-you-can-drive-my-car-how-we-eval.pdf>

CHEN, T ., HE, T ., BENESTY, M ., KHOTILOVICH, V ., TANG, Y ., CHO, H ., CHEN, K ., 2015. [online]. *Xgboost: extreme gradient boosting.* V : *R  package version 0 .4 -2*. [cit. 2022-03-03]. Dostupné na internete <https://cran.microsoft.com/snapshot/2017-12-11/web/packages/xgboost/vignettes/xgboost.pdf>

IBM Cloud Education, 2020. *What is machine learning?*. [online]. 2020 [cit. 2021-12-20] Dostupné na internete <https://www.ibm.com/cloud/learn/machine-learning>

Ketamo, H ., O'Rourke, P ., 2019. *Proceedings of Fake Intelligence Online Summit*. [online] Konferencia, 2019 [cit. 2022-01-01]. Dostupné na internete <https://www.theseus.fi/bitstream/handle/10024/169049/2019\_D \_1 \_SAMK\_Proceedings\_FakeIntelligenceOnlineSummit2019.pdf?sequence=1 >

Kramer, O ., 2016. *Machine learning for evolution strategies* (Vol. 20). Heidelberg: Springer, 2016. ISBN: 978-3 -319-33383-0

McCarthy, J ., 2007. *What is artificial intelligence?* [online]. Stanford, CA: Standford University, 2007. [cit. 2021-11-10]. Dostupné na internete <http://www-formal.stanford.edu/jmc/whatisai.pdf>

MCCORDUCK, P ., CFE, C ., 2004. *Machines who think: A  personal inquiry into the history and prospects of artificial intelligence*. [online]. CRC Press, 2004. [cit. 2022-12-02]. Dostupné na internete <https://monoskop.org/images/1 /1e/McCorduck\_Pamela\_Machines\_Who\_Think\_2nd\_ed.pdf>

MCKINNEY, W., 2011. *Pandas: a foundational Python library for data analysis and statistics*.[online]. V: *Python for high performance and scientific computing*. [cit. 2021-12-07]. pp.1-9. Dostupné na internete <https://www.dlr.de/sc/portaldata/15/resources/dokumente/pyhpc2011/submissions/pyhpc2011\_submission\_9.pdf>

MIJWEL, M .M ., 2018. Artificial neural networks advantages and disadvantages. [online]. University of Baghdad, 2018. [cit. 2021-12-02]. Dostupné na internete <https://www.researchgate.net/profile/Maad-Mijwil/publication/323665827\_Artificial\_Neural\_Networks\_Advantages\_and\_Disadvantages/links/5aa2c01faca272d448b5a23d/Artificial-Neural-Networks-Advantages-and-Disadvantages.pdf>

MOSCONI, F . 2019. Zero to Deep Learning. Fullstack.io, 2019. 769 s .

OWEN-HILL, A ., 2021. [online]. *5  Super-Dangerous Jobs That Robots Can Do Safely*.2021.[cit. 2022-02-04]. Dostupné na internete <https://blog.robotiq.com/5 -super-dangerous-jobs-that-robots-can-do-safely>

PATTERSON, D .W ., 2002. *Introduction to Artificial Intelligence and Expert Systems*. Prentice-Hall, Inc, 2002. 446 s . ISBN 81-203-G777-1

POOLA, I ., 2017. *How artificial intelligence in impacting real life everyday.* V : *International Journal for Advance Research and Development*.[online]. Panchkula: IJARnD. s . 96-100.[cit. 2021-12-20]. ID: V2I10-1170. Dostupné na internete < https://www.ijarnd.com/manuscripts/v2i10/V2I10-1170.pdf>

SACHDEV, H .S ., 2019. *Tensorflow vs Keras vs PyTorch vs Theano*. 2019. [cit. 2022-02-25]. Dostupné na internete < https://www.linkedin.com/pulse/tensorflow-vs-keras-pytorch-theano-harpreet-singh-sachdev?trk=portfolio\_article-card\_title>

SARAN, C ., 2019. *Stanford University finds that AI is outpacing Moore’s Law*. 2019. [cit. 2022-02-25] Dostupné na internete <https://www.computerweekly.com/news/252475371/Stanford-University-finds-that-AI-is-outpacing-Moores-Law#:~:text=Every%20three%20months%2C%20the%20speed,University%27s%202019%20AI%20Index%20report&text=Stanford%20University%27s%20AI%20Index%202019,AI)%20is%20outpacing%20Moore%27s%20Law.>

SHUKLA, N ., FRICKLAS, K ., 2018. [online] *Machine learning with TensorFlow*. Greenwich: Manning, 2018. [cit. 2022-01-11]. Dostupné na internete <http://manning-content.s3.amazonaws.com/download/3 /e0d6f80-038c-49b5-9c3d-1fd3c3bc9e4c/sample\_ch10\_Shukla\_Maching-Learning\_January23.pdf>

SMITH, P .D ., 2018. *Hands-On Artificial Intelligence for Beginners: An introduction to AI concepts, algorithms, and their implementation.* Packt Publishing, 2018. 362 s . ISBN 978-1788991063

SU, G ., 2018. *Unemployment in the AI Age*. V : *AI Matters*.[online]. New York: Association for Computing Machinery. s . 35-43.[cit. 2022-02-24]. DOI: 10.1145/3175502.3175511. Dostupné na internete <https://sigai.acm.org/static/aimatters/3 -4 /AIMatters-3 -4 -09-Su.pdf>

TURING, A ., 1950. Machinery and intelligence. Mind: V: *A  Quarterly Review of Psychology and Philosophy*. [online]. vol. 59, p .236. [cit. 2021-11-10]. DOI 10.1093/mind/LIX.236.433. Dostupné na internete <https://academic.oup.com/mind/article/LIX/236/433/986238>

WASKOM, M.L., 2021. *Seaborn: statistical data visualization*. V: *Journal of Open Source Software*. Vol: 6. i: 60, p.3021. [online]. [cit. 2022-01-2]. DOI: 10.21105/joss.03021. Dostupné na internete <https://joss.theoj.org/papers/10.21105/joss.03021.pdf>

WILMER, H .H ., SHERMAN, L .E ., CHEIN, J .M ., 2017. *Smartphones and cognition: A  review of research exploring the links between mobile technology habits and cognitive functioning.* V : *Frontiers in psychology*, Vol: 8 . p .605.[online]. Panchkula: IJARnD. s . 96-100.[cit. 2022-03-24]. DOI: [10.3389/fpsyg.2017.00605](https://doi.org/10.3389/fpsyg.2017.00605). Dostupné na internete <https://www.frontiersin.org/articles/10.3389/fpsyg.2017.00605/full>

YAN, Y., YAN, J., 2018*. Hands-on data science with Anaconda: utilize the right mix of tools to create high-performance data science applications.* Packt Publishing Ltd, 2018. 356 s. ISBN 978-1-78883-119-2

# Zoznam príloh

Príloha A  – ZIP súbor s  kurzom

PRÍLOHA A

**Príloha A**

* odkaz na GIT: https://github.com/batsysk/Umela\_inteligencia\_s\_pouzitim\_Pythonu
* súbor (Jupyter notebooks a obrázky): UI\_s\_použitím\_Pythonu\_KURZ.zip

1. https://www.google.com/webmasters/tools/legal-removal-request?complaint\_type=rtbf&visit\_id=637840908421470124-3072001186&rd=1  [↑](#footnote-ref-1)
2. https://edu.ukf.sk/pluginfile.php/112767/course/summary/bc\_ui.pdf [↑](#footnote-ref-2)
3. https://www.swi-prolog.org [↑](#footnote-ref-3)
4. priscilla.fitped.eu [↑](#footnote-ref-4)
5. https://www.muni.cz/o -univerzite/fakulty-a -pracoviste/fakulta-informatiky/331300-katstrojoveho-uceni-a -zpracovani-d /vyuka [↑](#footnote-ref-5)
6. https://is.muni.cz/predmet/fi/PV056?lang=cs;obdobi=5365 [↑](#footnote-ref-6)
7. https://is.muni.cz/predmet/fi/PA034?lang=en&obdobi=2362 [↑](#footnote-ref-7)
8. https://is.muni.cz/predmet/phil/jaro2022/PLIN068 [↑](#footnote-ref-8)
9. https://www.vut.cz/studenti/predmety/detail/99241?apid=99241&aid\_redir=1  [↑](#footnote-ref-9)
10. https://www.python.org [↑](#footnote-ref-10)
11. Zdroj Obrázok1: <https://statisticstimes.com/tech/top-computer-languages.php> (printscreen) [↑](#footnote-ref-11)
12. https://www.springboard.com/blog/data-science/best-language-for-machine-learning/ [↑](#footnote-ref-12)
13. https://scikit-learn.org/stable [↑](#footnote-ref-13)
14. https://www.tensorflow.org [↑](#footnote-ref-14)
15. https://xgboost.readthedocs.io/en/latest [↑](#footnote-ref-15)
16. https://keras.io [↑](#footnote-ref-16)
17. https://numpy.org [↑](#footnote-ref-17)
18. theano-pymc.readthedocs.io/en/latest [↑](#footnote-ref-18)
19. https://pytorch.org [↑](#footnote-ref-19)
20. https://pandas.pydata.org/docs [↑](#footnote-ref-20)
21. https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html [↑](#footnote-ref-21)
22. Zdroj Obrázok 2 : autor (printscreen) [↑](#footnote-ref-22)
23. https://seaborn.pydata.org [↑](#footnote-ref-23)
24. https://matplotlib.org [↑](#footnote-ref-24)
25. https://docs.anaconda.com [↑](#footnote-ref-25)
26. https://jupyter.org [↑](#footnote-ref-26)
27. Zdroj Obrázok 3 : <https://priscilla.fitped.eu/categories> (printscreen) [↑](#footnote-ref-27)
28. Zdroj Obrázok 4 : autor (printscreen) [↑](#footnote-ref-28)
29. Zdroj Obrázok 5 : autor (printscreen) [↑](#footnote-ref-29)
30. Zdroj Obrázok 6 : autor (printscreen) [↑](#footnote-ref-30)
31. Zdroj Obrázok 7 : autor (printscreen) [↑](#footnote-ref-31)
32. Zdroj Obrázok 8 : autor (printscreen) [↑](#footnote-ref-32)
33. https://colab.research.google.com/notebooks/welcome.ipynb?hl=sk [↑](#footnote-ref-33)
34. Zdroj Obrázok 9 : autor (printscreen) [↑](#footnote-ref-34)
35. Zdroj Obrázok 10: autor (printscreen) [↑](#footnote-ref-35)
36. Zdroj Obrázok 11: autor (printscreen) [↑](#footnote-ref-36)
37. Zdroj Obrázok 12: autor (printscreen) [↑](#footnote-ref-37)
38. Zdroj Obrázok 13: autor (printscreen) [↑](#footnote-ref-38)
39. Zdroj Obrázok 14: autor (printscreen) [↑](#footnote-ref-39)
40. Zdroj Obrázok 15: autor (printscreen) [↑](#footnote-ref-40)
41. Zdroj Obrázok 16: autor (printscreen) [↑](#footnote-ref-41)
42. Zdroj Obrázok 17: autor (printscreen) [↑](#footnote-ref-42)
43. Zdroj Obrázok 18: autor (printscreen) [↑](#footnote-ref-43)
44. Zdroj Obrázok 19: autor (printscreen) [↑](#footnote-ref-44)
45. Zdroj Obrázok 20: autor (printscreen) [↑](#footnote-ref-45)