Tiriamoji duomenų analizė

# EDA – Exploratory data analysis – 120 valandų kursas susiskirstytas į 30 paskaitų po 4 valandas

Kursą sudaro

1. Jupyter notebook
2. Numpy
3. Tiesinė algebra
4. Tikimybių teorija
5. Statistika
6. Pandas
7. Vizualizavimas
   1. Kaip vizualizuoti
   2. Matplotlib
   3. Seaborn
8. Exploratory data analysis – teorija
9. Exploratory data analysis – praktika

Kurso medžiaga išdalinta į 30 paskaitų po 4 valandas

# Reikalavimai

1. Git
2. Virtual environment
3. Python

# 1 Paskaita – notebook + numpy

## Jupyter notebook

1. Instaliavimas – pip install jupyter
2. Globalūs nustatymai - ~/.jupyter
3. Lokalūs nustatymai -
4. Klaviatūros komandos – esc+h
5. Cell tipas – code, markdown
   1. Markdown cheatsheet <https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet>
6. Code cell formatavimas
   1. Black
      1. <https://github.com/psf/black>
      2. https://hackernoon.com/using-black-to-auto-format-your-python-8cu338f
7. Papildomi plėtiniai
   1. <https://towardsdatascience.com/jupyter-notebook-extensions-517fa69d2231>
8. Cell width (cell plotis)
   1. <https://stackoverflow.com/questions/21971449/how-do-i-increase-the-cell-width-of-the-jupyter-ipython-notebook-in-my-browser>

## Numpy

1. Intro - <https://numpy.org/devdocs/user/absolute_beginners.html>
   1. Kas yra numpy?
   2. Numpy vs python list
   3. Absolute basics for beginners

# 2 Paskaita – numpy

## Numpy

1. Array creation
   1. <https://numpy.org/devdocs/user/basics.creation.html>
2. Indexing
   1. <https://numpy.org/devdocs/user/basics.indexing.html>
3. Data types
   1. <https://numpy.org/devdocs/user/basics.types.html>
4. Broadcasting
   1. <https://numpy.org/devdocs/user/basics.broadcasting.html>

# 3 Paskaita – tiesinė algebra

## Tiesinė algebra

## 1 variantas

Knyga - Thomas Nield - Essential Math for Data Science.

Nuoroda į knygos skyrių – Tiesinė algebra  
<https://learning.oreilly.com/library/view/essential-math-for/9781098102920/ch04.html#idm44994589512736>

Nuoroda į knygos github - <https://github.com/thomasnield/oreilly_math_fundamentals_data_science>

Skaidrė yra mūsų direktorijoje

\notebooks\tiesine\_algebra\1 variantas - knygos material \math\_fundamentals\_linear\_algebra.pdf

## 2 variantas

Material repozitorija   
<https://github.com/pabloinsente/math-app-ml>

Idėtas notebook‘as

notebooks\tiesine\_algebra\2 varianas\notebooks\01-linear-algebra-white-background.ipynb

Idėtas pdf

notebooks\tiesine\_algebra\2 varianas\pdf\ linear\_algebra.pdf

## Papildomos užduotys klasėje abiems variantams

Impelementuoti tiesinės algebros veiksmus su python list duomenų tipu.

1. Vekoriu sudėtis
2. Matricos transponavimas
3. Matricų daugyba
4. Dot product
5. Determinanto skaičiavimas
6. Tiesinių lygčių sistemos sprendimas

# 4 Paskaita – Praktika – numpy

# 5 Paskaita - tiesinė algebra + matematinė analizė ir funkcijos

Numpy ir tiesinė algebra – pavyzdys kaip naudojama tiesinė algebra

<https://numpy.org/numpy-tutorials/content/tutorial-svd.html>

Jeigu tiesinės algebros teorija išdėstyta – dėstyti calculus and functions. Svarbu padengti

1. Kas yra funkcija
2. Kas yra išvestinė

Funkcijos ir išvestinės konceptai naudojami Machine Learning algoritmuose.

## Papildoma užduotis klasėje

1. Implementuoti Gradient descent algoritmą naudojantis python list duomenų tipu.

# 6 Paskaita – algoritmai

* Rekursija
* Greedy algoritmai
* Brute force

1. Rekursija <https://www.geeksforgeeks.org/recursion/>
   1. Užduotis – sukurti funkciją, kuri grąžina Fibonačio skaičių. F(n) = n-tasis Fibonači skaičius.
   2. Keys and values žodyne
2. Greedy algoritmai
   1. Vagies problema - <https://www.geeksforgeeks.org/fractional-knapsack-problem/>
   2. Auksakasio problema - <https://www.geeksforgeeks.org/gold-mine-problem/>
3. Apeiti šachmatų lentą su žirgu - <https://www.chess.com/terms/knights-tour-chess#solve-knights-tour-on-chesscom>

# 7 Paskaita – algoritmai

1. Tree structures – binary tree <https://www.geeksforgeeks.org/binary-tree-set-1-introduction/>
   1. Taikymo pavyzdžiai – Random Forest algoritmas.
   2. Užduotis 1. sukurti tree struktūrą
   3. Užduotis 2. Sukurti algoritmą, kuris grąžina visas medžio viršūnes
   4. Užduotis 3. Panaudoti užduoties 2 sprendimą viršūnių paieškai. Pateikiamas viršūnės numeris ir funkcija grąžina viršūnę, arba None, jeigu tokios viršūnės nėra.
2. Grafai - <https://www.geeksforgeeks.org/graph-data-structure-and-algorithms/>
   1. Trumpiausio kelio algoritmas <https://www.geeksforgeeks.org/dijkstras-shortest-path-algorithm-greedy-algo-7/?ref=lbp>

Užduotis: implementuoti trumpiausio kelio algoritmą.

* 1. Depth first search - <https://www.geeksforgeeks.org/depth-first-search-or-dfs-for-a-graph/>
  2. Breadth first search - <https://www.geeksforgeeks.org/breadth-first-search-or-bfs-for-a-graph/?ref=lbp>
  3. Užduotis 1. Implementuoti Depth first search algoritmą
  4. Užduotis 2. Implementuoti Breadth first search algoritmą

# 8 Paskaita – Praktika – numpy