

Projekt z manažmentu dát

Contents

0 Úvod	3
0.1 Použité dáta	3
1 Spracovanie dát	4
1.1 Konverzia	4
1.2 Ďalšie spracovanie	4
2 Výpočet a uloženie dát	5
2.1 Idea	5
2.2 Použité algoritmy	5
2.3 Použité metriky	5
3 Predstavenie stránky	7
4 Porovnanie kombinácií s najväčšou úspešnosťou	9
5 Záver	11

0 Úvod

Cieľom nášho projektu bolo vytvoriť Flask aplikáciu, v ktorej užívateľ môže porovnať rôzne binárne klasifikačné algoritmy na daných datasetoch spôsobom ľubovoľného výberu premenných. Na porovnanie slúžia rôzne metriky, ktoré rozoberieme neskôr. Takisto cieľom bolo preskúmať, ktoré premenné považoval daný algoritmus za „najdôležitejšie“.

0.1 Použité dáta

Ako dáta sme použili 4 medicínske datasety – 2 ohľadom klasifikácie diabetu a 2 ohľadom klasifikácie kardiovaskulárnych ochorení. Každý dataset je jedinečný a obsahuje rozdielne medicínske merania a rôzny počet sledovaných jednotiek. Každý obsahuje binárnu premennú: „Diagnosis“, či u pacienta bolo potvrdený výskyt diabetu, infarktu myokardu alebo iného kardiovaskulárneho ochorenia. Túto premennú budeme predpovedať našimi algoritmami.

1 Spracovanie dát

1.1 Konverzia

Prvým krokom bolo zjednotiť formáty daných súborov. 3 zo 4 boli vo formáte .csv, ktorý bol pre našu úlohu vyhovujúci, preto sme aj zvolili ukladanie vstupných dát do takýchto súborov. Posledný dátový súbor sme získali vo formáte .arff a preto sme sa ho snažili prekonvertovať pomocou python-u do .csv. Na toto slúži skript `to_csv.py`, ktorého outputom je čistý dataset obsahujúci len merania a názvy premenných, bez ostatného, prebytočného textu.

1.2 Ďalšie spracovanie

Vo všeobecnosti, na narábanie s datasetmi sme použili knižnicu `pandas`.

Ďalším krokom bolo zjednotiť názov a hodnoty sledovanej premennej a tak isto prekonvertovať kategoriálne premenné typu *string* na typ *int*.

Pred spracovaním týchto datasetov sa naša sledovaná premenná vyskytovala v rôznych podobách. Napr. rozdielne názvy ako: „target“, „diabetes“ a taktiež hodnoty týchto premenných boli „nuly“ a „jednotky“, alebo „positive“ a „negative“, preto sme za vhodný krok zvolili premenovať všetky tieto sledované premenné na „*Diagnosis*“ a ako hodnoty voliť len „nula“ alebo „jedna“.

Pre potrebu daných algoritmov sme kategoriálne premenné typu „*Gender*“ s hodnotami „male“, „female“ skonvertovali takisto na „0 alebo 1“.

Toto jednoduché predspracovanie dát sa nachádza v súbore „*change.py*“.

2 Výpočet a uloženie dát

2.1 Idea

Pre potrebu aplikácie, teda možnosť si zvoliť ľubovoľnú kombináciu premenných pre daný dataset a preskúmať výsledky metrík sme všetky tieto informácie pred počítali a uložili do formátu .json.

2.2 Použité algoritmy

Ako binárne klasifikátory sme použili tieto algoritmy:

- Logistickú regresiu
- Náhodný les (Random Forest)
- Gaussian Naive Bayes
- Rozhodovací strom (Decision Tree)
- KNN (K-Nearest Neighbours)

Všetky tieto algoritmy sme importovali z knižnice sci-kit learn.

2.3 Použité metriky

- „*Accuracy score*“
 - je metrika, ktorá vyjadruje pomer správnych predpovedí (teda správne klasifikovaných prípadov) k celkovému počtu prípadov. Vzorec na výpočet presnosti je nasledovný:

$$\text{Presnosť} = \frac{\text{Počet správnych predpovedí}}{\text{Celkový počet predpovedí}}$$

- „*Precision score*“
 - je metrika, ktorá meria schopnosť modelu správne identifikovať pozitívne prípady z tých, ktoré predpovedal ako pozitívne. Vyjadruje sa nasledovne:

$$\text{Presnosť} = \frac{TP}{TP + FP}$$

kde: TP (true positive) označuje skutočné pozitívne prípady správne predpovedané modelom

FP (false positive) sú negatívne prípady nesprávne predpovedané ako pozitívne

- „*Recall score*“
 - Citlivosť (recall) je metrika, ktorá meria schopnosť modelu správne identifikovať všetky skutočné pozitívne prípady v dátach. Vyjadruje sa nasledovne:

$$\text{Citlivosť} = \frac{TP}{TP + FN}$$

kde: TP (true positive) sú skutočné pozitívne prípady správne predpovedané modelom

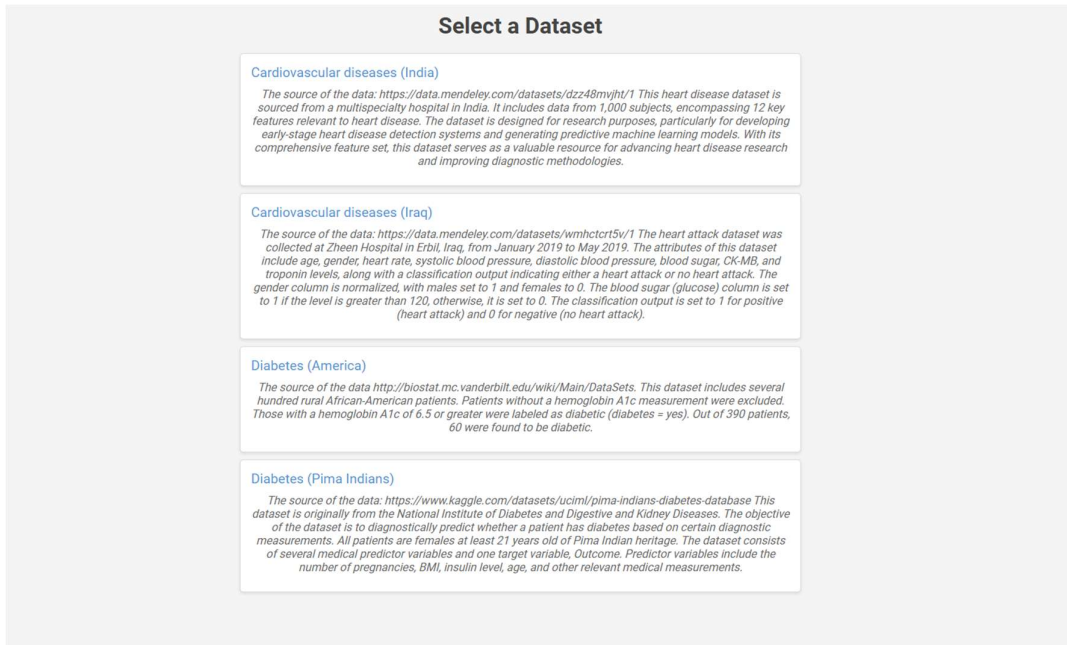
FN (false negative) sú pozitívne prípady nesprávne predpovedané ako negatívne.

- „F1 score“
 - je metrika, ktorá kombinuje presnosť (precision) a citlivosť (recall) do jedného čísla. Je to harmonický priemer týchto dvoch metrík a poskytuje vyvážené hodnotenie, najmä keď je dôležité zohľadniť ako falošné pozitívne, tak aj falošné negatívne prípady

$$F1 = 2 * \frac{Presnosť * Citlivosť}{Presnosť + Citlivosť}$$

- „ROC AUC score“
 - ROC AUC skóre (Receiver Operating Characteristic - Area Under Curve) je metrika, ktorá meria schopnosť modelu odlišovať medzi pozitívnymi a negatívnymi triedami. ROC krivka je grafické znázornenie vzťahu medzi citlivosťou (recall) a špecifickosťou (1 - false positive rate) pri rôznych prahoch rozhodovania modelu.
 - Hodnoty ROC AUC skóre:
 - 0.9 - 1.0: Vynikajúca schopnosť rozlišovať medzi triedami.
 - 0.8 - 0.9: Veľmi dobrá schopnosť rozlišovať medzi triedami.
 - 0.7 - 0.8: Dobrá schopnosť rozlišovať medzi triedami.
 - 0.6 - 0.7: Priemerná schopnosť rozlišovať medzi triedami.
 - 0.5 - 0.6: Slabá schopnosť rozlišovať medzi triedami.

3 Predstavenie stránky



Select a Dataset

Cardiovascular diseases (India)
The source of the data: <https://data.mendeley.com/datasets/dzz48mmyjt/1> This heart disease dataset is sourced from a multispecialty hospital in India. It includes data from 1,000 subjects, encompassing 12 key features relevant to heart disease. The dataset is designed for research purposes, particularly for developing early-stage heart disease detection systems and generating predictive machine learning models. With its comprehensive feature set, this dataset serves as a valuable resource for advancing heart disease research and improving diagnostic methodologies.

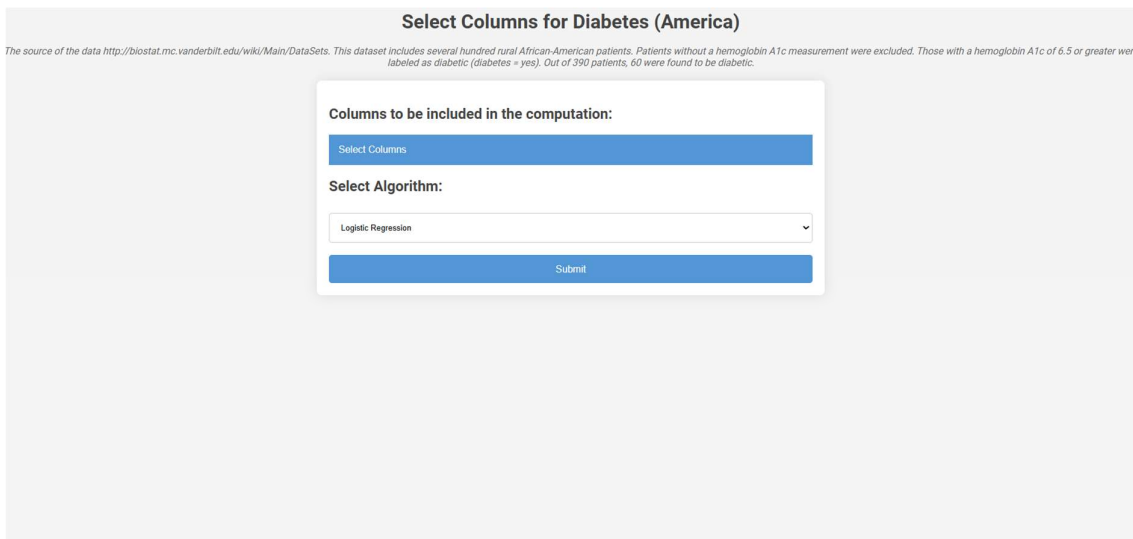
Cardiovascular diseases (Iraq)
The source of the data: <https://data.mendeley.com/datasets/wmhctcr15v/1> The heart attack dataset was collected at Zheen Hospital in Erbil, Iraq, from January 2019 to May 2019. The attributes of this dataset include age, gender, heart rate, systolic blood pressure, diastolic blood pressure, blood sugar, CK-MB, and troponin levels, along with a classification output indicating either a heart attack or no heart attack. The gender column is normalized, with males set to 1 and females to 0. The blood sugar (glucose) column is set to 1 if the level is greater than 120, otherwise, it is set to 0. The classification output is set to 1 for positive (heart attack) and 0 for negative (no heart attack).

Diabetes (America)
The source of the data <http://biostat.mc.vanderbilt.edu/wiki/Main/DataSets>. This dataset includes several hundred rural African-American patients. Patients without a hemoglobin A1c measurement were excluded. Those with a hemoglobin A1c of 6.5 or greater were labeled as diabetic (diabetes = yes). Out of 390 patients, 60 were found to be diabetic.

Diabetes (Pima Indians)
The source of the data: <https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database> This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective of the dataset is to diagnostically predict whether a patient has diabetes based on certain diagnostic measurements. All patients are females at least 21 years old of Pima Indian heritage. The dataset consists of several medical predictor variables and one target variable, Outcome. Predictor variables include the number of pregnancies, BMI, insulin level, age, and other relevant medical measurements.

Obrázok 1

Úvodná stránka s možnosťou výberu datasetu, na ktorom chceme vykonávať porovnanie.



Select Columns for Diabetes (America)

The source of the data <http://biostat.mc.vanderbilt.edu/wiki/Main/DataSets>. This dataset includes several hundred rural African-American patients. Patients without a hemoglobin A1c measurement were excluded. Those with a hemoglobin A1c of 6.5 or greater were labeled as diabetic (diabetes = yes). Out of 390 patients, 60 were found to be diabetic.

Columns to be included in the computation:

Select Columns

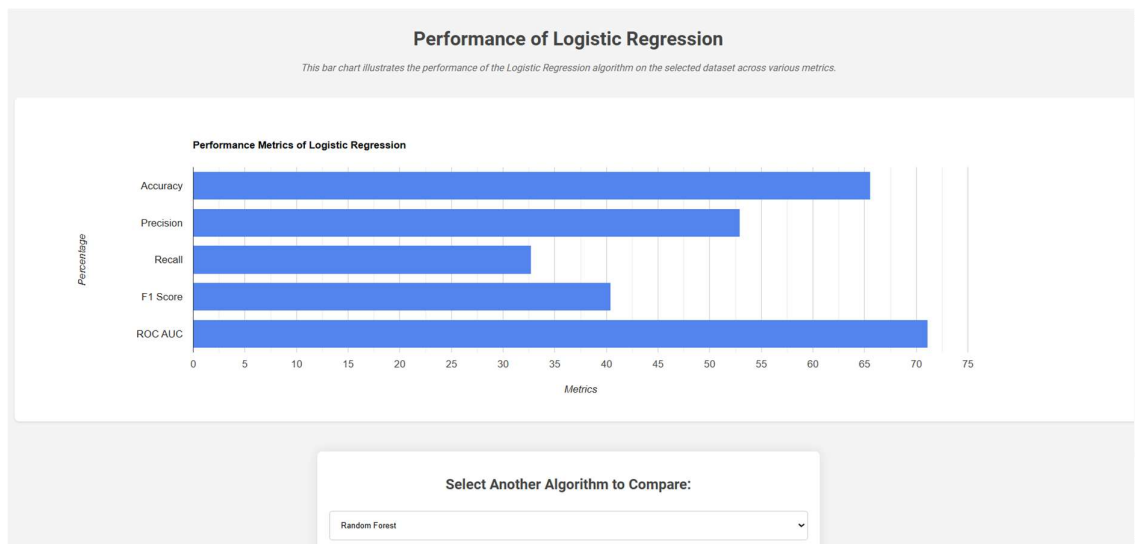
Select Algorithm:

Logistic Regression

Submit

Obrázok 2

Po vybratí datasetu si môžeme vybrať ktoré premenné (stĺpce) zahrnieme do výpočtu daného algoritmu, ktorý si taktiež vyberáme.



Obrázok 3

Po vybratí premenných a algoritmu môžeme vidieť bar chart, ktorý zobrazuje jednotlivé metriky pre daný výber. Tento výber môžeme porovnať s ľubovoľným iným algoritmom.



Obrázok 4

Finálna stránka, kde môžeme vidieť porovnanie dvoch algoritmov s daným výberom premenných pomocou „stacked bar chart“

4 Porovnanie kombinácii s najväčšou úspešnosťou

Dataset	Algorithm	Avg Score	Combination
Cardiovascular diseases (India)	DecisionTreeClassifier	0,986083237	Fasting_blood_sugar, Gender, Max_heart_rate, Num_of_major_vessels, Oldpeak, Resting_electrocardiogram, Serum_cholesterol, Slope
Cardiovascular diseases (India)	GaussianNB	0,988420792	Exercise_angina, Gender, Max_heart_rate, Resting_electrocardiogram, Serum_cholesterol, Slope
Cardiovascular diseases (India)	KNeighborsClassifier	0,977260908	Resting_electrocardiogram, Slope
Cardiovascular diseases (India)	LogisticRegression	0,988355167	Age, Exercise_angina, Fasting_blood_sugar, Max_heart_rate, Oldpeak, Resting_blood_pressure, Resting_electrocardiogram, Slope
Cardiovascular diseases (India)	RandomForestClassifier	1	Age, Chest_pain, Fasting_blood_sugar, Num_of_major_vessels, Resting_blood_pressure, Resting_electrocardiogram, Serum_cholesterol, Slope
Diabetes (Pima Indians)	DecisionTreeClassifier	0,86092521	Chol_hdl_ratio, Cholesterol, Gender, Glucose, Hips, Systolic_blood_press
Diabetes (Pima Indians)	GaussianNB	0,897752795	BMI, Chol_hdl_ratio, Glucose, Waist_hip_ratio
Diabetes (Pima Indians)	KNeighborsClassifier	0,869090532	Glucose, HDL-cholesterol, Systolic_blood_press
Diabetes (Pima Indians)	LogisticRegression	0,907687138	Cholesterol, Glucose, Height, Hips, Waist_hip_ratio
Diabetes (Pima Indians)	RandomForestClassifier	0,887610565	BMI, Cholesterol, Glucose, Height, Hips, Systolic_blood_press, Waist_hip_ratio
Diabetes (America)	DecisionTreeClassifier	0,748217411	Age, BMI, Diabetes_pedigree, Plasma_glucose, Serum_insuling, Skin_thickness
Diabetes (America)	GaussianNB	0,787736034	BMI, Diabetes_pedigree, Plasma_glucose, Pregnancy, Serum_insuling
Diabetes (America)	KNeighborsClassifier	0,708141152	Age, BMI, Plasma_glucose, Pregnancy
Diabetes (America)	LogisticRegression	0,742368713	Diabetes_pedigree, Plasma_glucose, Pregnancy, Skin_thickness
Diabetes (America)	RandomForestClassifier	0,747853468	Age, BMI, Diabetes_pedigree, Diastolic_pressure, Plasma_glucose, Pregnancy
Cardiovascular diseases (Iraq)	DecisionTreeClassifier	0,983164392	Creatine_Kinase, Diastolic_blood_pressure, Heart_rate, Systolic_blood_pressure, Troponin
Cardiovascular diseases (Iraq)	GaussianNB	0,837743098	Age, Creatine_Kinase, Systolic_blood_pressure, Troponin
Cardiovascular diseases (Iraq)	KNeighborsClassifier	0,894816509	Creatine_Kinase, Troponin
Cardiovascular diseases (Iraq)	LogisticRegression	0,848162638	Age, Creatine_Kinase, Diastolic_blood_pressure, Heart_rate, Systolic_blood_pressure, Troponin
Cardiovascular diseases (Iraq)	RandomForestClassifier	0,988125622	Creatine_Kinase, Diastolic_blood_pressure, Gender, Heart_rate, Systolic_blood_pressure, Troponin

Tabuľka 1

Tabuľka zobrazuje pre najúspešnejšie kombinácie premenných pre datasety podľa každého algoritmu.

„Avg score“ označuje priemer všetkých použitých metrík.

Na tabuľke môžeme vidieť, že úspešnosť daných algoritmov závisí aj na vybranom datasete.

Z tejto analýzy sa zdá ako najúspešnejší „Random Forest“. Avšak, pri behu všetkých algoritmov bral aj najviac času. Predpokladáme, že to vychádza z povahy tohto algoritmu, keďže vytvára viackrát Decision Tree.

Takisto si môžeme všimnúť, že algoritmus KNN má pre maximálne skóre menší počet premenných a teda aj menej informácie. Avšak ani so všetkými premennými nedokázal so svojim skóre „predbehnúť“ ostatné porovnávané algoritmy.

Ďalšou zaujímavosťou z našej analýzy bolo vyvrátenie nášho predpokladu, že algoritmy budú mať tendenciu vyberať si rovnaké premenné. Predpokladali sme, že premenné, ktoré budú „niesť“ informáciu o datasete budú jednotné. Z Tabuľky 1 nie je vidieť akási štruktúra výberu takýchto premenných. Otázne zostáva, či to je spôsobené malým počtom porovnávaných algoritmov alebo iným faktorom.

Priemerné skóre pre algoritmy z tabuľky 1:

- Logistic Regression: 0.871
- Random Forest: 0.905
- Naive Bayes: 0.877
- Decision Tree: 0.894

5 Záver

Jednou z ťažších vecí pre nás určite bolo spojazdnenie Flask aplikácie, pretože sme ani jeden poriadne nerozumeli ako sa spracovávajú requesty, ktoré stránka dostáva.

Na druhú stranu tvorenie grafov v javascripte aj pre nás bolo rozumne rýchle, vďaka dokumentácii, ktorá nám bola sprístupnená počas prednášok.

Zaujímavou časťou určite boli napríklad spoznanie samotných metrík, ktoré sa používajú pri vyhodnocovaní binárnych klasifikátorov. Cennou skúsenosťou bolo aj „ohmatanie“ si používania týchto algoritmov z knižnice sci-kit learn, i keď im zatiaľ po matematickej stránke nerozumieme dopodrobna.

Delba práce:

Tomáš Antal

- Report
- app.py
- change.py
- index.html
- statistics.ipynb
- algorithm.html

Tomáš Varga

- Protokol
- to_csv.py
- algorithms.py
- dataset.html
- compare.html
- styles.css – ChatBot generované (nami len mierne upravené)

PS:

Na koniec by sme chceli povedať, že práca jedného z nás bola dopĺňaná prácou druhého. Okrem práce na reporte a protokole sme spolu výrazne spolupracovali.