Пловдивски университет „Паисий Хилендарски”

**Факултет по математика и информатика**

**Катедра *Компютърна информатика***

**ДИПЛОМНА РАБОТА**

**Прогнозиране на успеха на студенти чрез използване на методи за машинно обучение**

**Дипломант: Михаил Йорданов Йорданов**

**Фак. №: 1801322019**

**Научен ръководител: доц. д-р Силвия Гафтанджиева**

**Пловдив**

**2022 г.**

**Съдържание**

**TODO:**

**(hyperlinked content viewer)**

1. УВОД +
2. МАШИННО ОБУЧЕНИЕ +
   * 1. ДЕФИНИЦИЯ +
     2. ОБЩ МОДЕЛ НА МАШИННОТО ОБУЧЕНИЕ +
     3. РЕГРЕСИОНЕН АНАЛИЗ +
3. МЕТОДИ +
   * 1. ИЗБРАНИ МЕТОДИ +
        1. GRADIENT BOOSTED Regressor
        2. RANDOM FOREST Regressor
        3. EXTRA TREES Regressor
        4. POLYNOMIAL REGRESSION +
        5. LINEAR REGRESSION +
        6. ELASTICNET REGRESSION
4. МАТЕРИАЛИ
5. ПРИЛОЖЕНИЕ И РЕЗУЛТАТИ
6. ОЦЕНКА
   * 1. СРАВНЕНИЕ НА МЕТОДИ
7. ЗАКЛЮЧЕНИЕ
8. СПИСЪК НА СЪКРАЩЕНИЯТА
9. ЛИТЕРАТУРА

**1. Увод**

Бързото развитие и широкото разпространение на компютърните системи и Интернет водят до разрастването на публично достъпни данни, които да бъдат анализирани. Тези данни могат да бъдат под формата на всякакъв вид информация за определена дейност или статистика, като например трафик на уеб сайт, потребителски предпочитания и навици или поръчки. Основният проблем често е размерът на данните, тъй като те са твърде големи, за да бъдат анализирани от хора. Компютрите от друга страна могат да обработват големи количества от данни много бързо.

Концепцията за машинно обучение се поражда от тази обстановка, като нейната идея е “изкуствено” генериране на знания от опит: системата се учи от примери и може да ги обобщи след завършване на етапа на обучение.

Машинното обучение намира приложение в доста сфери от нашето ежедневие. Например при онлайн търсачките като Google, Bing, DuckDuckGo и много други, се използва машинно обучение за осъществяване на по-добра връзка между търсените фрази и показаните като резултат уеб страници. Анализирайки съдържанието на уеб страниците, търсачките могат да дефинират кои думи и фрази са най-значителни при дефинирането на конкретна уеб страница. Използвайки това връщат най-уместните резултати за дадена търсена фраза.

В зависимост от вида на данните, алгоритмите за машинно обучение се разделят на два вида – контролирано (supervised) и безконтролно (unsupervised) машинно обучение. При контролираното обучение ние “учим” модела и след това с тези знания той може да прогнозира непознати или бъдещи случаи. Много важен отличителен белег на контролираното обучение е, че данните са белязани (labeled). От своя страна контролираното обучение се дели на 2 вида: Класификация и Регресия. Като Класификацията е процес, при който се прогнозират дискретни категории или означения, а при Регресията се прогнозират непрекъснати стойности.

***Тематиката на дипломната работа*** е свързана със сравняването на различни методи за машинно обучение, които прогнозират крайния успех на студенти. Оценява се точността на методите, както и потенциалната грешка в прогнозата.

**2. Машинно обучение**

2.1. Дефиниция

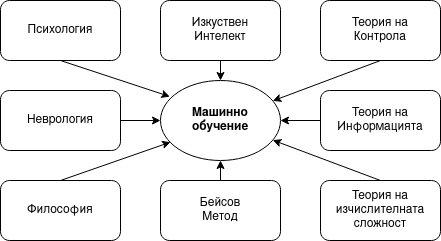
Ученето е процес на придобиване или модифициране на вече съществуващи стойности, познания, предпочитания или поведения. Машините, за разлика от хората, зависят от данни: учат се от опит. На фундаментално ниво машинното обучение е категория от изкуствения интелект, която позволява на компютрите да “мислят” и да се самообучават. По този начин те променят действията си, с цел подобряване на точността, която се измерва от гледна точка на това колко пъти взетото решение е довело до правилния резултат.

“Казва се, че компютърната програма се учи от опит *E* със зависимост от клас от задачи *T* и производителност *P*, ако производителността за извършването на задачите *T*, както е измерена от *P*, се увеличава с опита *E*.”

* Том М. Мичъл

С други думи, машинното обучение е възможността на компютъра да се учи от опит, който обикновено е даден във формата на входни данни. Гледайки данните, той може да намери зависимости в тях, които са твърде сложни, за да бъдат намерени от човек. Машинното обучение може да се използва за откриване на скрити структури от класове в неструктурирани данни или за откриване на зависимости в структурирани данни, за да направи прогноза.

Машинното обучение е мултидисциплинарно поле, имайки широк обхват от изследователски области, подпомагащи неговото развитие (фиг. 1).



*фиг. 1*

2.2 Общ модел на машинното обучение

Машинното обучение се използва за решаването на проблеми, чрез самообучение от страна на машината. Такива проблеми имат 3 особености:

* Клас от задачи
* Подобряване на производителността
* Процесът на трупане на опит

Общият модел на машинното обучение се състои от 6 компонента, независещи от възприетия алгоритъм (фиг. 2).



*фиг. 2*

Всеки компонент има за цел да изпълни конкретна задача:

1. **Събиране и подготовка на данни:** Основната задача в този процес е да се съберат и подготвят данните във формат, в който могат да се подадат като вход на алгоритъма.
2. **Избор на функции:** Изчистват се ненужни и повтарящи се данни, също така се преработват някои стойности, като например се заместват NaN стойностите с 0.
3. **Избор на алгоритъм:** При избора на алгоритъм за машинно обучение трябва първо да се определи целта на задачата. Конкретни алгоритми са по-подходящи за решаването на дадена задача от други.
4. **Избор на модел и параметри:** Повечето алгоритми за машинно обучение изискват първоначална намеса в избора на най-подходящите стойности на параметрите.
5. **Трениране:** След избора на алгоритъм и параметрични стойности, моделът трябва да се тренира, използвайки част от данните като training data.
6. **Оценка на производителността:** Преди да се имплементира в дадена система, моделът трябва да се тества върху непознати за него данни, за да се определи неговата точност.

2.3 Видове машинно обучение

Контролираното обучение, безконтролното обучение и подсиленото обучение са три основни концепции на машинното обучение.

2.3.1 Контролирано обучение

При контролираното обучение входните данни са поднесени като етикетирани. Моделът може да се учи от тях и да предостави резултат на дадения проблем.

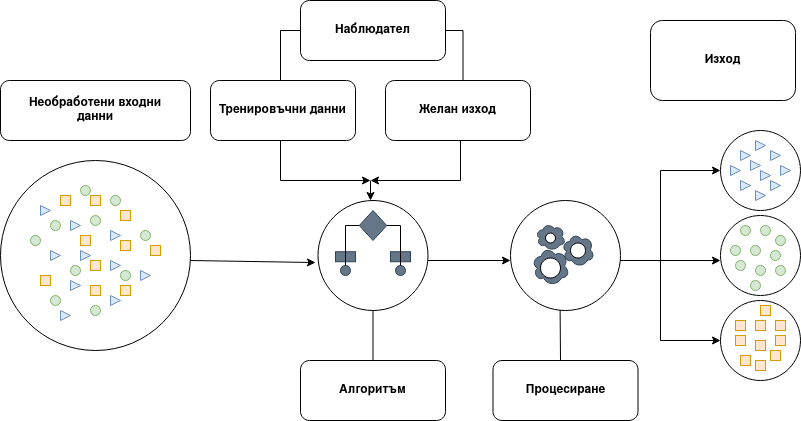
**Видове алгоритми:**

Контролираното обучение има два вида алгоритми – класификация и регресия. Класификацията е процес на класифициране на етикетираните данни. Моделът създава граници, които разделят категориите данни. Примери са:

* Naive Bayes Classifier
* Support Vector Machines
* Logistic Regression

Регресията е процесът на прогнозиране на тенденцията на предишните данни за предсказване на резултата от новите данни. Изходът може да се състои от една или повече непрекъснати променливи. Прогнозирането се извършва с помощта на линия, която покрива повечето точки от данните. Примери са:

* Linear Regression
* Nonlinear Regression
* Bayesian Linear Regression

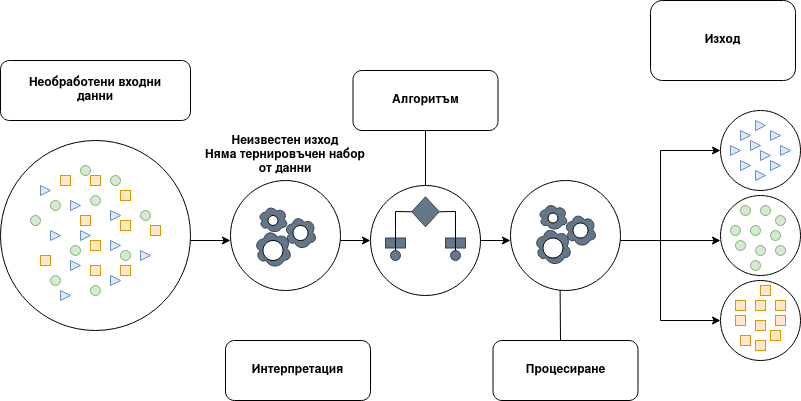


*фиг. 3*

2.3.2 Безконтролно обучение

При безконтролното обучение моделът работи самостоятелно. Той използва алгоритми за машинно обучение, за да стигне до заключения относно неетикирани данни. Основната му цел е да проучи скритите модели на данните и да прогнозира изхода. Примерни алгоритми са:

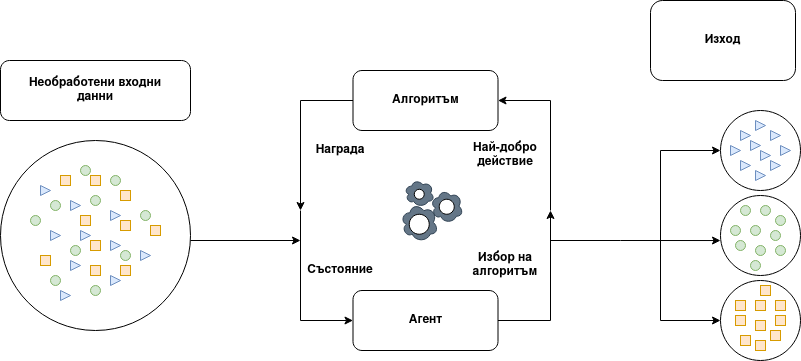
* K-Means clustering
* Neural Networks
* Principal Component Analysis



*фиг. 4*

2.3.3 Подсилено обучение

При подсиленото обучение винаги има начално и крайно състояние. За да достигне крайното състояние, агентът (learning agent) манипулира средата, като съответно получава награда (appreciation) при успех и не получава нищо при провал. По този начин агентът се учи от средата.



*фиг. 5*

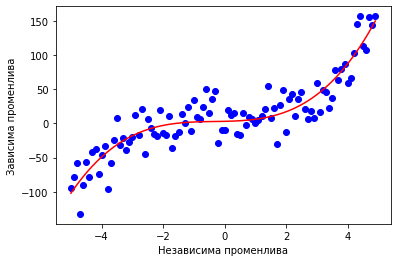
2.3.4 Ключови разлики между трите вида обучение

* Контролираното обучение обхваща регресия и класификация. Безконтролното обучение се занимава с групиране (clustering) и търсене на асоциативни (associative) връзки. Докато подсиленото обучение използва изучаване на средата, в която се намира.
* Контролираното обучение използва етикетирани данни и изходните данни са познати на системата. Безконтролното обучение използва неетикетирани данни, където изхода се базира на колекцията от възприятията. Подсиленото обучение използва средата с дискретни стъпки, там данните са предефинирани.
* Целта на контролираното обучение е да генерира формула, базирана на входните и изходните данни. При безконтролното обучение се намира асоциация между входните данни и се групират. При подсиленото обучение агентът се учи чрез пряко взаимодействие със средата.

|  |  |  |  |
| --- | --- | --- | --- |
| **Критерий** | **Контролирано обучение** | **Безконтролно обучение** | **Подсилено обучение** |
| Дефиниция | *Учи се използвайки етикетирани данни* | *Учи се без надзор, използвайки неетикетирани данни* | *Учи се взаимодействайки със средата* |
| Видове данни | *Етикетирани данни* | *Неетикетирани данни* | *Предефинирани данни* |
| Видове задачи | *Регресия и класификация* | *Асоциация и групиране* | *Експлоатация или разучаване* |
| Алгоритми | *Linear Regression, Logistic Regression, SVM, KNN и др.* | *K-Means, C-Means, Apriori и др.* | *Q-Learning, SARSA и др.* |
| Цел | *Изчисляване на резултатите* | *Откриване на скрити възприятия* | *Научаване на поредица от действия* |
| Приложение | *Оценка на риска, Прогнозиране на продажби* | *Системи за препоръки, Откриване на аномалии* | *Самостоятелно управлявани автомобили, Игри, Здравеопазване* |

2.4 Регресионен анализ

Регресионен анализ е форма на техника за прогнозен анализ, която разглежда взаимоотношението между зависима и независими променливи. Тази техника се използва за прогнозиране, моделиране на времеви серии и откриване на причинно-следствени връзки между променливите.



Регресионният анализ е важен инструмент за моделиране и анализиране на данни. Напасва се крива/права върху точките от данните по начин, такъв че да се минимизират разликите между разстоянията от точките.

**3. Методи**

Както е описано по-подробно в точка 2.3, алгоритъмът на машинното обучение е метод, който системата използва, за да изпълни задачата, която сме й задали. Калифорнийският университет – Бъркли разделя системата за учене на алгоритмите за машинно обучение на три основни части:

1. **Процес на вземане на решения**:

В общи линии, алгоритмите за машинно обучение се използват, за да се направи прогноза или класификация. Базирано на входните данни, който могат да са етикетирани или неетикетирани, алгоритъмът ще произведе приблизителна оценка свързана с модела на данните.

2. **Функция на грешките**:

Функцията на грешките се използва за оценяване на прогнозата на модела. При наличието на познати примери, функцията на грешките може да направи сравнение, за да оцени точността на модела.

3. **Процес на оптимизация на модела**:

Ако моделът може да се напасне по-добре към данните в тренировъчния набор, следователно тежестта трябва да се регулира, за да се намали несъответствието между познатите примери и прогнозираното от алгоритъма. Алгоритъмът повтаря този процес на оценка и оптимизация, като актуализира тежестта докато не достигне прага на точността.

3.1 Избрани методи

3.1.1 Gradient Boosted Regression

**Усилване на Градиента** (Gradient Boosting) е един от вариантите на ансамбълните методи, където се създават множество от “слаби” модели и се обединяват за повишаване на точността и производителността. Този вид регресия има следната структура:

1. Инициализиране на модела със константна стойност:

където:

– начална стойност на прогнозата

*L* – функция на загубата

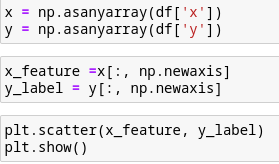
*argmin* – означава, че търсим стойност на , за която се минимализира загубата *L*

2. Крайна форма на модела:

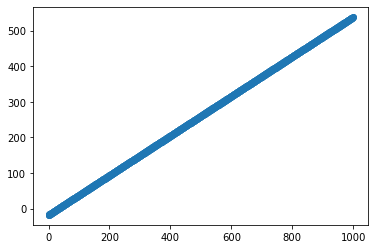
[https://towardsdatascience.com/all-you-need-to-know-about-gradient-boosting-algorithm-part-1-regression-2520a34a502]

**Примерна дистрибуция на данни:**

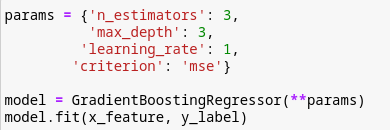
*Определяне и форматиране на данни*



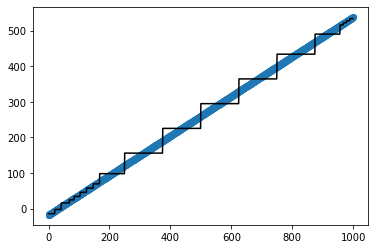
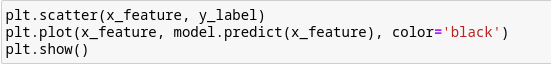
*Визуализиране на данните чрез pyplot*



*Прогнозиране на данни използвайки* ***Gradient Boosted Regression***



*Визуализация на модела*

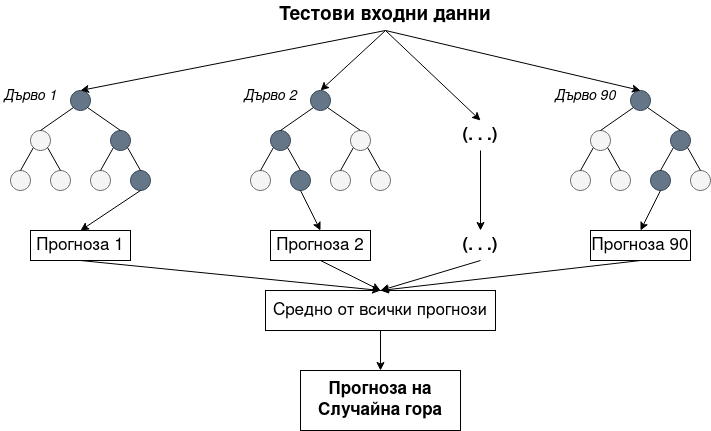


*Виждаме визуализация на функцията, минаваща през точките на данните и съставните й по-малки функции.*

3.1.2 Random Forest

**Случайна гора** е ансамбълна техника както за регресия, така и за класификация, използваща множество дървета на решенията и техника наречена Bootstrap and Aggregation, позната още като bagging (пакетиране). Този метод комбинира резултатите от всяка прогноза, като обединява множеството от дървета на решенията. Този алгориъм има следната структура:

[https://medium.com/swlh/random-forest-and-its-implementation-71824ced454f]



*За всяко дърво на решението се използва следната формула:*

*където:*

– *важността на разклонение j*

– *притегленият брой проби достигнали разклонение j*

– *замърсената стойност на разклонение j*

*left(j)* – *разклонение-дете от лявото деление на разклонение j*

*right(j)* –*разклонение-дете от дясното деление на разклонение j*

*Важността на всяко свойство на дърво на решението се изчислява по този начин:*

*където:*

– *важността на свойство i*

*След това се нормализират в стойности от 0 до 1, като се раздели на сумата от всички стойности на важността на свойствата. След което се изчислява крайната важност на свойствата, като се изчисли средното между всички дървета.*

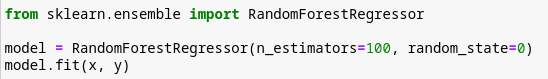
[https://towardsdatascience.com/the-mathematics-of-decision-trees-random-forest-and-feature-importance-in-scikit-learn-and-spark-f2861df67e3]

**Примерна дистрибуция на данни:**

*Определяне и форматиране на данни:*



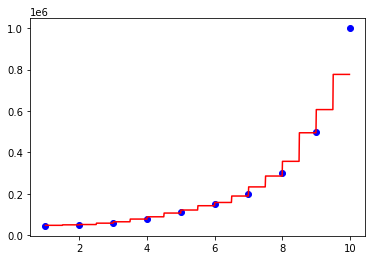
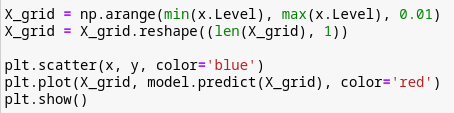
*Напасване на данните върху модела:*



*Прогнозиране на данни, използвайки модела:*



*Визуализация на модела:*



*Виждаме визуализация на функцията, минаваща през точките на данните.*

[https://www.geeksforgeeks.org/random-forest-regression-in-python/]

3.1.3 Extra Trees

**Extra Trees Регресията** (Extremely Randomized Trees) е сходна с горепосочената Случайна Гора. И двете са съставени от голям брой дървета на решенията, където за крайния изход се вземат предвид прогнозите на всяко дърво. Но докато Случайната Гора избира оптималния клон, Extra Trees избира произволен.

[https://quantdare.com/what-is-the-difference-between-extra-trees-and-random-forest/]

*Формулата за изчисление на ентропия е:*

*където:*

*c – е броят на уникалните класови етикети*

*– е пропорцията от редове с изходни етикети i*

[https://www.geeksforgeeks.org/ml-extra-tree-classifier-for-feature-selection/]

**Примерна дистрибуция на данни:**

*Разделяне на данните в тренировъчни и тестови*



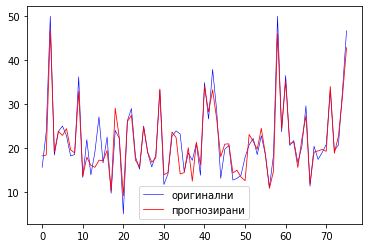
*Напасване на данните върху модела*



*Прогнозиране на данни върху модела*



*Визуализация на модела*



*Виждаме точността на прогнозата съпоставена с оригиналните данни.*

[https://www.datatechnotes.com/2020/06/regression-example-with-extratrees-in-python.html]

3.1.4 Multiple Linear Regression

**Множествената Линейна Регресия** е техника от статистиката, използваща няколко независими променливи, за да прогнозира изхода на една зависима променлива. Този вид регресия има следната структура:

където:

– *променлива на резултата (response variable)*

– *брой функции (number of features)*

– *n-та функция (n-th feature)*

– *коефициент на регресията (тежест) на n-та функция*

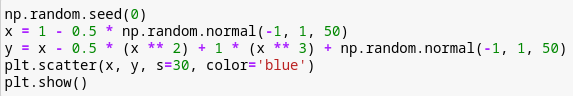
*(regression coefficient (weight) of the n-th feature)*

- *пресечна точка (intercept)*

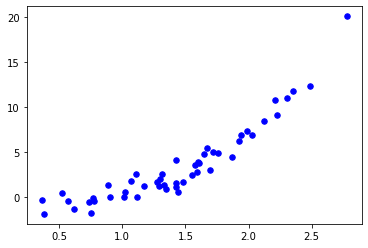
*[https://www.investopedia.com/terms/m/mlr.asp]*

**Примерна дистрибуция на данни:**

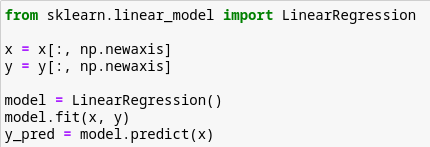
*Генериране на произволни данни*



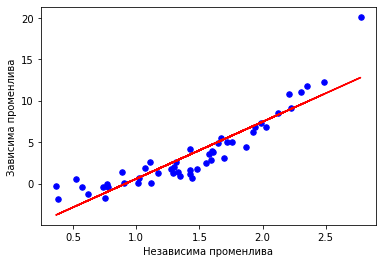
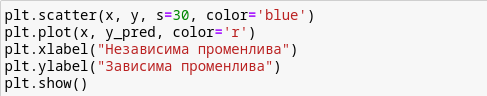
*Визуализиране на данните чрез pyplot*



*Прогнозиране на данни използвайки Линейна Регресия*



*Визуализация на модела*



*Виждаме линейната визуализация на функцията, минаваща през точките на данните.*

3.1.5 Polynomial Regression

В **Полиномната Регресия** нелинейната връзка между независимата променлива x и зависимата y, е описана като многочлен от n-та степен. В зависимост от сложността на данните, степента на многочлена се покачва, но при много висока степен може да се получи свръх оборудване на данните, което да доведе до висока стойност на грешката при тестване. Този вид регресия има следната структура:

*където:*

– *променлива на резултата (response variable)*

– *брой функции (number of features)*

– *n-та степен на функцията (n-th degree feature)*

– *коефициент на регресията (тежест) на n-та функция*

*(regression coefficient (weight) of the n-th feature)*

- *пресечна точка (intercept)*

За всяка точка от данните могат да се направят изчисления със следната формула:

От което се формира следната матрица:

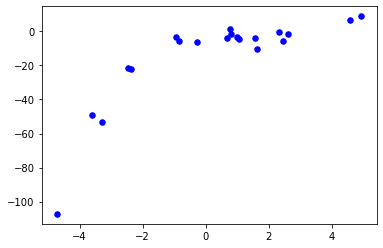
=  +

**Примерна дистрибуция на данни:**

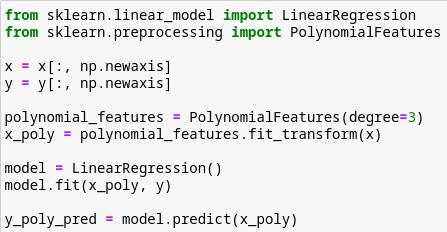
*Генериране на произволни данни*



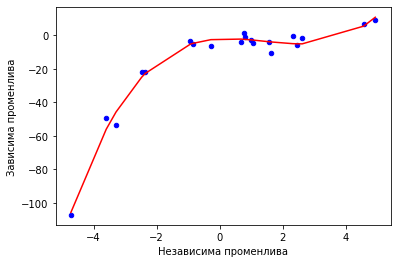
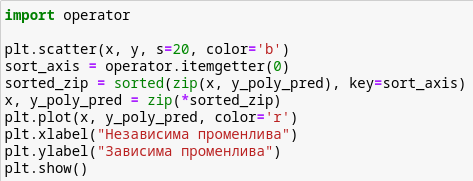
*Визуализиране на данните чрез pyplot*



*Прогнозиране на данни използвайки Полиномна Регресия*



*Визуализация на модела*



*Виждаме визуализация на кривата на функцията, минаваща през точките на данните.*

3.1.6 ElasticNet Regression

**ElasticNet Регресията** е регуларизиран регресионен метод, който работи с избиране и регулиране на променливи едновременно. Регуляризацията се изпълнява, когато моделът работи добре с тренировъчните данни, но се представя зле с данните за тестване. Регуляризацията е техника, която намалява грешките като напасва функция върху тренировъчните данни. Тези функции се наричат наказателни.

Има два вида наказателни функции  *и . Модел използващ*  се нарича Ласо Регресия (Lasso Regression), докато модел използващ се нарича Било регресия (Ridge Regression)*.* ElasticNet комбинира и двата метода.

*Формула за Lasso Regression:*

*Формула за Ridge Regression:*

*Формула за ElasticNet Regression:*

**Примерна дистрибуция на данни:**

*Разделяне на данните на тренировъчни и тестови*



*Инициализиране и напасване на модела върху данните*

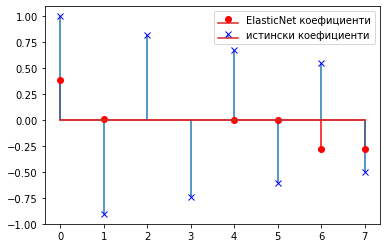
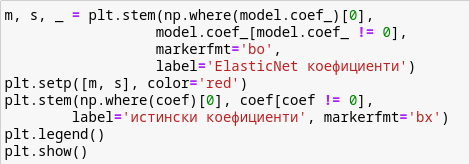


*Прогнозиране на данни*



*Изчисляване на коефициента на модела*

*Визуализация на модела*



*Виждаме прогнозираните коефициенти на модела и реалните коефициенти в тестовите данни.*

[https://analyticsindiamag.com/hands-on-tutorial-on-elasticnet-regression/]

**4. Материали TODO**

Текст

**5. Приложение и резултати TODO**

Текст

**6. Оценка TODO**

Коефициентът на детерминация () е статистически показател, използван за измерването на това колко от вариацията в резултата може да се обясни с вариацията в независимите променливи. винаги расте при добавянето на нови зависими променливи към модела, дори те да нямат връзка с изходната променлива.

Коефициентът може да има стойност само между 0 и 1, като при 0 означава, че изходът не може да бъде прогнозиран, от която и да е от променливите, а при 1 означава, че изходът се прогнозира без наличието на грешка.

**7. Заключение TODO**

Текст

**8. Списък на съкращенията TODO**

Текст

**9. Литература TODO**

Текст