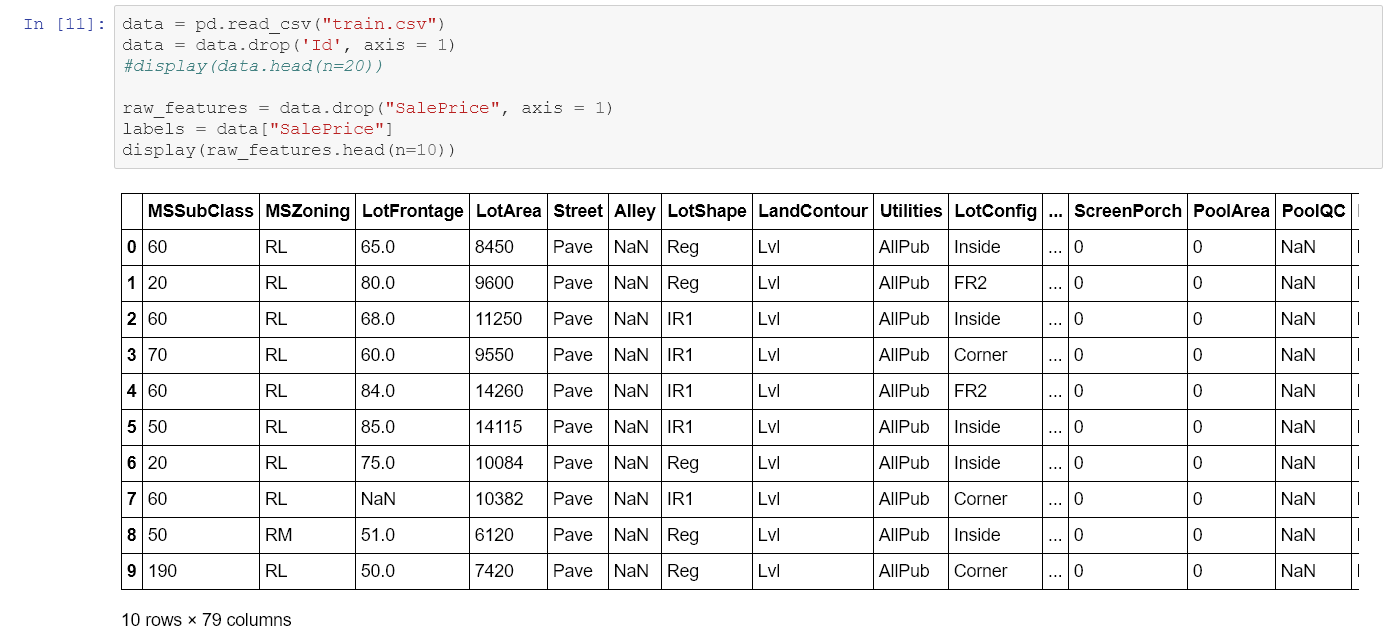
Machine Learning Final Project

Idrive Global

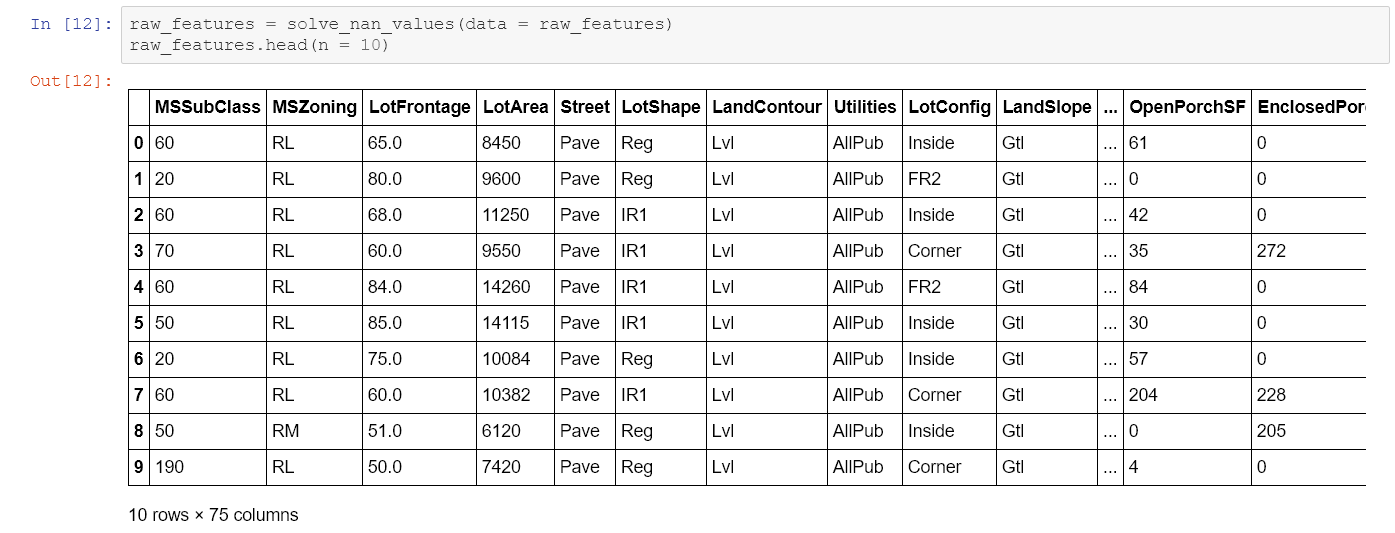
Autor: Adrian Iordache

Primul Pas… Incarcarea si Vizualizarea Datelor

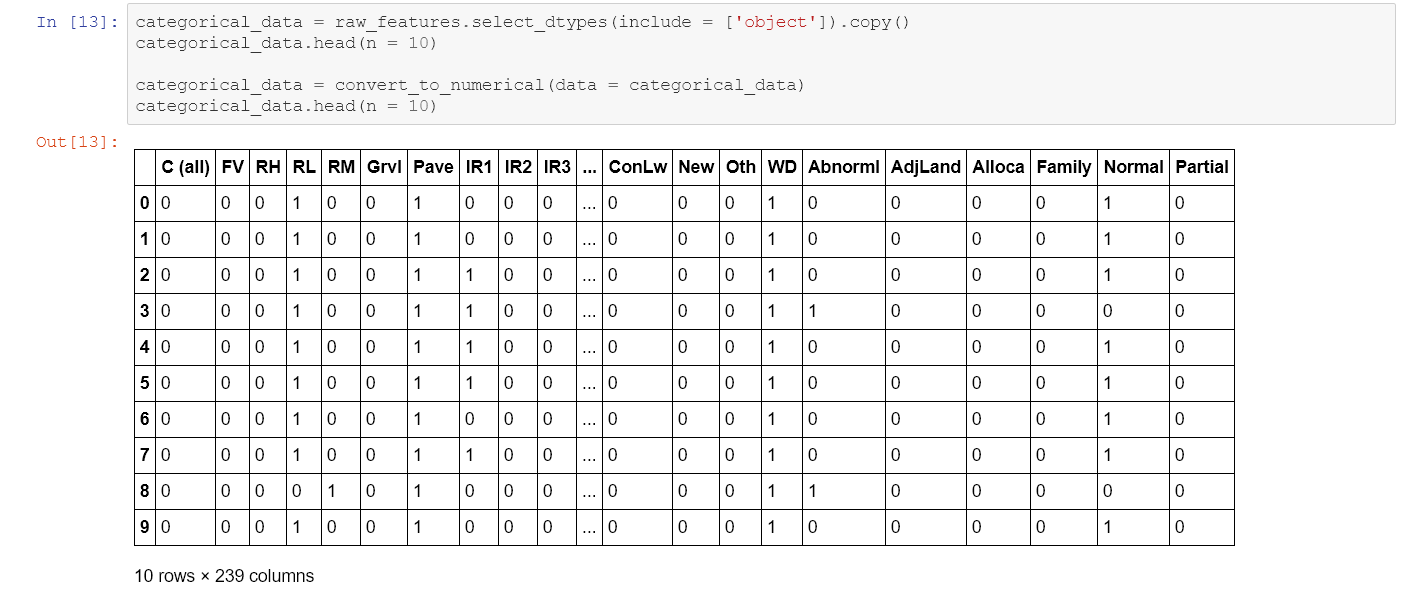


Se poate observa faptul ca datasetul contine valori NA (Non-Available), asa ca in primele faze este necesara o curatare a datelor.

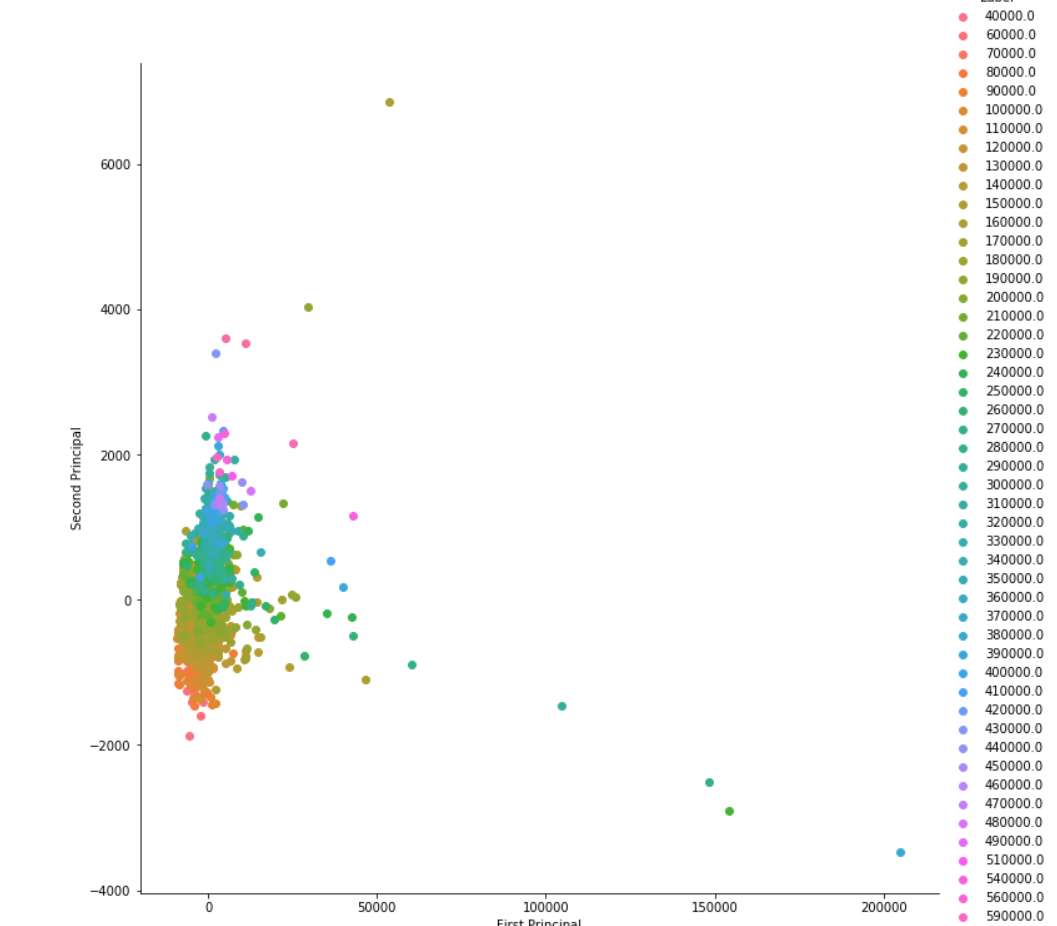
Vom face asta prin eliminarea acelor coloane daca numarul de elemente NA din aceasta este peste 75% din numarul de total sau prin inlocuirea valorii NA cu valoarea cea mai frecventa din coloana respectiva.



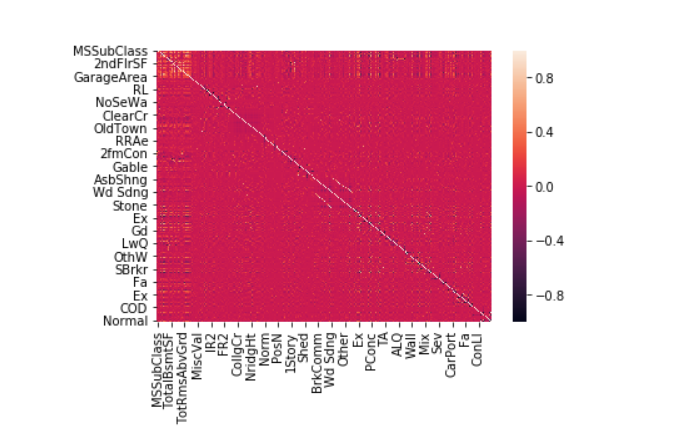
Dupa curatarea datelor, urmeaza pasul de convertire a datelor de tip non-numerical in date de tip numerical.



Vizualizarea datelor obtiunute pe un grafic ce reduce dimensionalitatea intr-un cadru 2D prin metoda PCA (Principal Component Analysis).

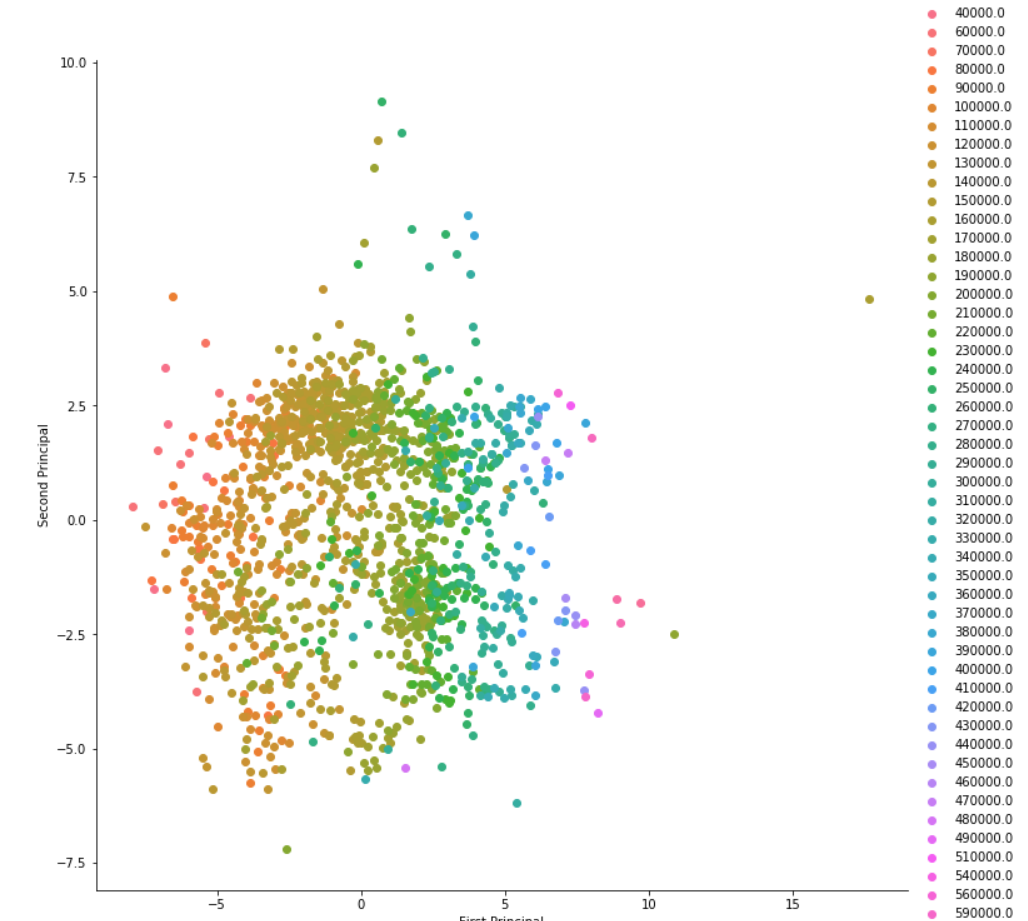


Observandu-se numarul mare de dimensiuni al datasetului obtinut vom incerca prin nivelul de corelatie intre coloane sa vedem daca putem renunta la un numar de dimensiuni pentru o mai buna eroare a modelului.



Dupa incercare mai multor metode de preprocesare precum Batch Normalization, MinMaxScaler, MaxAbsScaler, sklearn.preprocessing, Binarizer, Normalizer, StandardScaler.

S-a ajuns la concluzia ca cea mai eficienta metoda de preprocesare este StandardScaler cu eliminarea unor dimensiuni ce dispun de un indice de corelatie mai mare de 90%.



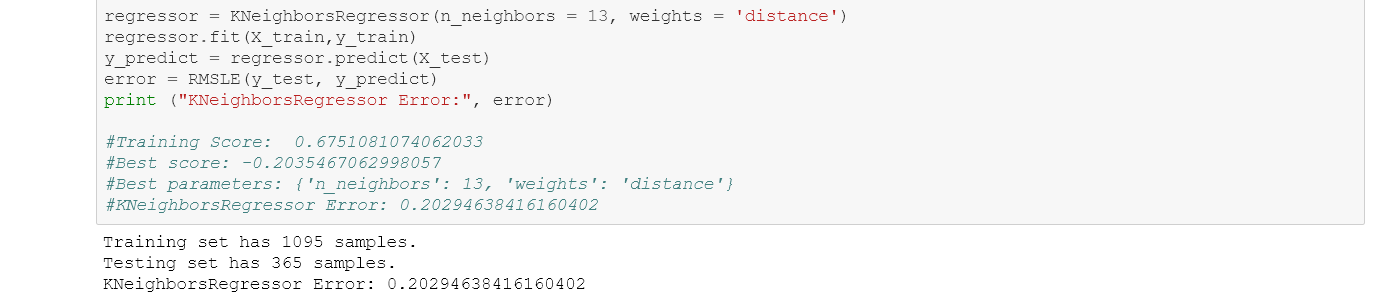
Dupa acesti pasi de preprocesare a datelor puntem incepe antrenarea unor modele in cautare unei erori mai mica de 23%.

Primul model antrenat pentru o eroare initiala estimativa a fost un

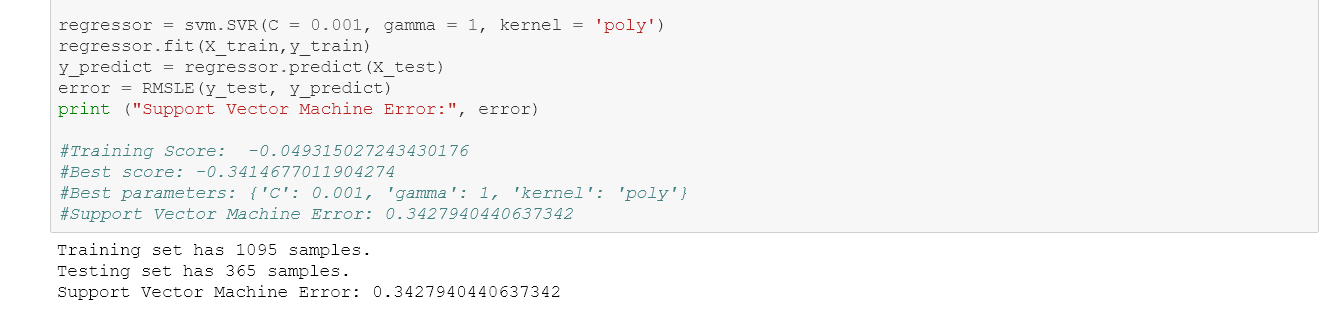
RandomForestRegressor neparametrizat ce a obtinut o eroare de 38%.

Pe parcursul sesiunilor de antrenamente cu GridSearchCV si CrossValidation pentru evitarea posibilei situatii de overfit s-au folosit urmatorii algoritmi de Machine Learning cautandu-se minimizarea functiei de eroare RMSLE (Root Mean Squared Logarithmic Error).

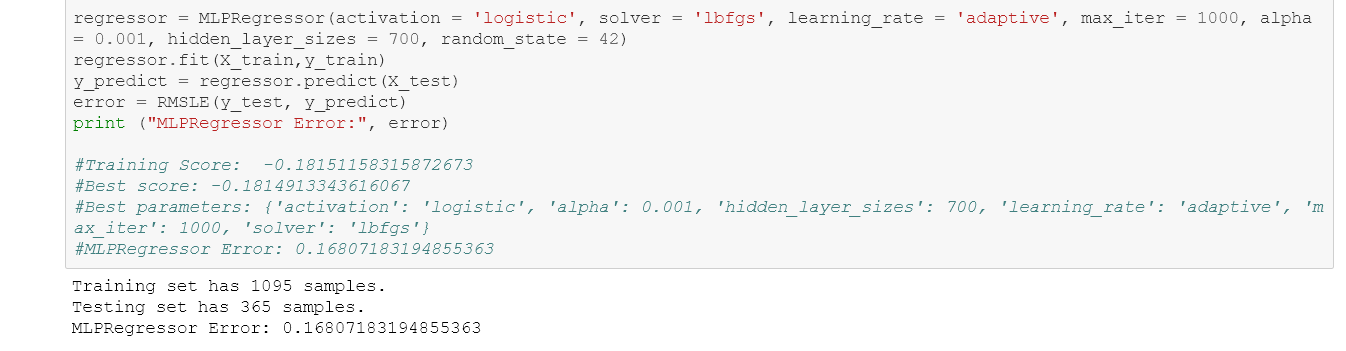
1. KNN Regressor (K-Nearest Neighbors)



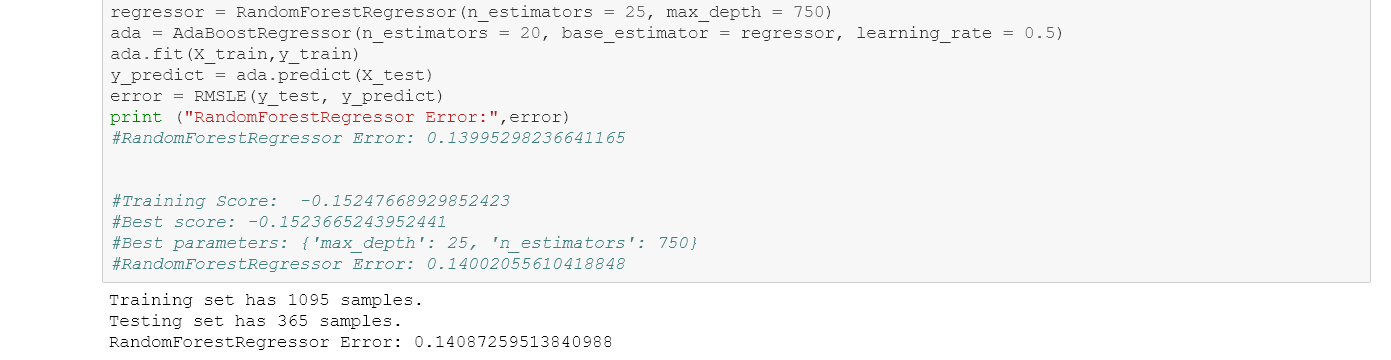
1. Support Vector Machine



1. MLPRegressor (Multilayer-Perceptron)



Fiecare dintre acestia a obtinut o eroare decenta, insa algoritmul care a adus rezultatul final a fost un AdaBoostRegressor parametrizat ce foloseste ca estimator un RandomForestRegressor cu parametri gasiti prin hiper tunare cu GridSearchCV.



Limita de eroare ce trebuia depasita era de 23%, iar modelul antrenat ajungand la o eroare de 14% denota finalizarea cu success a acestui proiect.