QMBU 450 - Final Project

Regression Analysis

- Abdul Rahman Hamadeh
- Mouafak Alsaid Hasan

Description:

Regression Analysis is a project where we tested three different regression functions on three different data sets and compared the overall performance of the regression functions.

The regression functions being: Linear regression, polynomial regression and nonparametric regression.

Data Retrieval:

We used three different data sets which are: Real estate, insurance and car price data sets retrieved from Kaggle website as `csv` files. We then imported them to our workspace in `python`.

The data sets are located in a folder called *Datasets* in the project files.

Feature Manipulation:

- **Feature extraction:**

  Using the CHI2 function from SKlearn library we picked the most impactful features on the Y values of the data set and did the modeling according to them.

- **Dimensionality Reduction:**

We used PCA from Sklearn library to perform the dimensionality reduction on the remaining features. For all the data sets, we reduced the dimensionality to 1 in order to be able to plot the regression function in 2 dimensions so we could have better interpretation on the data.

Hyperparameters Optimization:

We had 2 hyperparameters to optimize.

- The first one is the degree of the polynomial of the Polynomial Regression. The optimization was a trial over 40 values to find the degree of the polynomial with minimum RMSE.
- The second one is the window size of the parzan windows in the Nonparametric Regression.
  This hyperparameter was optimized with RMSE value: the model was tested with multiple *h* (window size) values and the *h* that gave the least RMSE value was picked as the optimal *h*.

Equations and Formulas:

- **Nonparametric Regression:**
  For the kernel smoother we used the following formula for the kernel function:

$$\text{Where } K(u) = \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}$$

And for the estimator:

$$\widehat{y}_i = \frac{\sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right) y_i}{\sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right)}$$

- **Linear Regression:**

  The formula of Linear Regression is:

$$Y = f(x) + \epsilon$$

  Where ϵ is a zero-mean Gausian distribution noise.

  A function g(x) = $w_1 x + w_0$ is an approximation for $f(x)$.

  Minimizing mean-square error, we can derive a formula for the parametersw_1, w_0.

$$w = A^{-1} Y$$

  where : $x_i$point in the dataset, *N* the number of data points

$$A = \begin{bmatrix} N & \sum_i x_i \\ \sum_i x_i & \sum_i (x_i)^2 \end{bmatrix}$$

$$Y = \begin{bmatrix} \sum_i y_i \\ \sum_i y_i x_i \end{bmatrix}$$

- **Polynomial Regression:**

  For K-degree polynomial, we have K +1 parameters: $w_0, w_1, \ldots w_{K-1}, w_K$,where

  $\mathbf{g(x \mid w_0, w_1, \ldots w_{K-1}, w_K) = w_0 + w_1 x + w_2 (x)^2 + \ldots + w_K (x)^K}$

  The formula to estimate the value of the parameters is:

$$\boldsymbol{w} = (\boldsymbol{D}^T\boldsymbol{D})^{-1}\boldsymbol{D}^T\boldsymbol{y}$$

D is the feature matrix scales with a vector of ones.

y is the vector of labels.

$$\mathbf{D} - \begin{bmatrix} 1 & x^1 & (x^1)^2 & \cdots & (x^1)^k \\ 1 & x^2 & (x^2)^2 & \cdots & (x^2)^k \\ \vdots & & & & \\ 1 & x^N & (x^N)^2 & \cdots & (x^N)^k \end{bmatrix}$$

- **The Root Mean Squared Error:**

$$\sqrt{\frac{\sum_{i=1}^{N_{test}}\left(y_i - \hat{y}_i\right)^2}{N_{test}}}$$

- **The Mean Squared Error:**

$$\frac{\sum_{i=1}^{N_{test}}\left(y_i - \hat{y}_i\right)^2}{N_{test}}$$

Findings and Conclusion:

We observed that for data sets with a well-defined trend (as in real estate and car price data sets) all three regression functions performed relatively well.

We also observed that the regression model's accuracy is affected by the trend of the data and the features selected from the data.

Findings:

We see that the Real estate dataset is more suitable for polynomial regression. The insurance dataset has no trend, hence we observe that the models don't fit with any regression model and leads to high RMSE. The Car-price dataset is more suitable for polynomial dataset. The high RMSE of this dataset is because of the high values of the labels, which seems logical in this case.

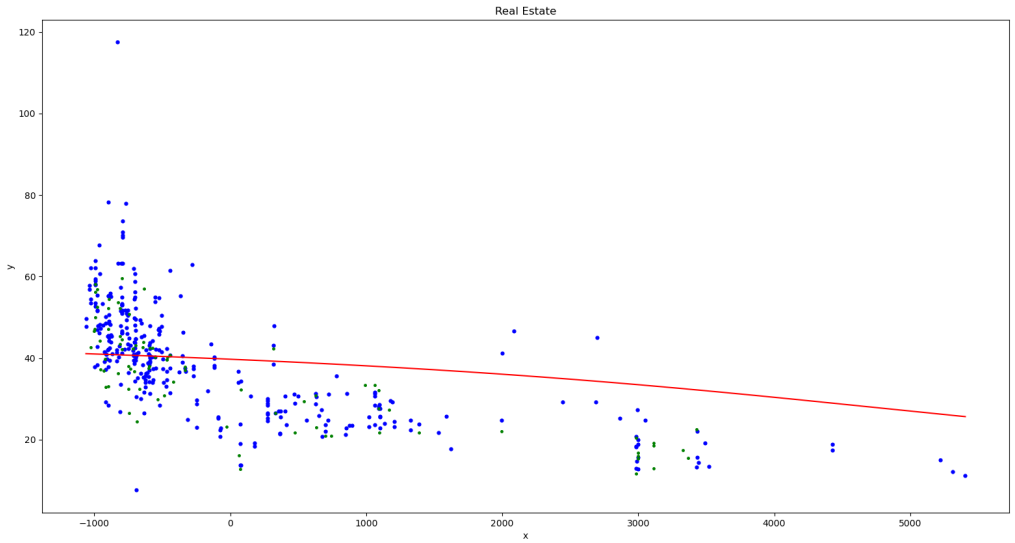
Figures:

Nonparametric Regression:

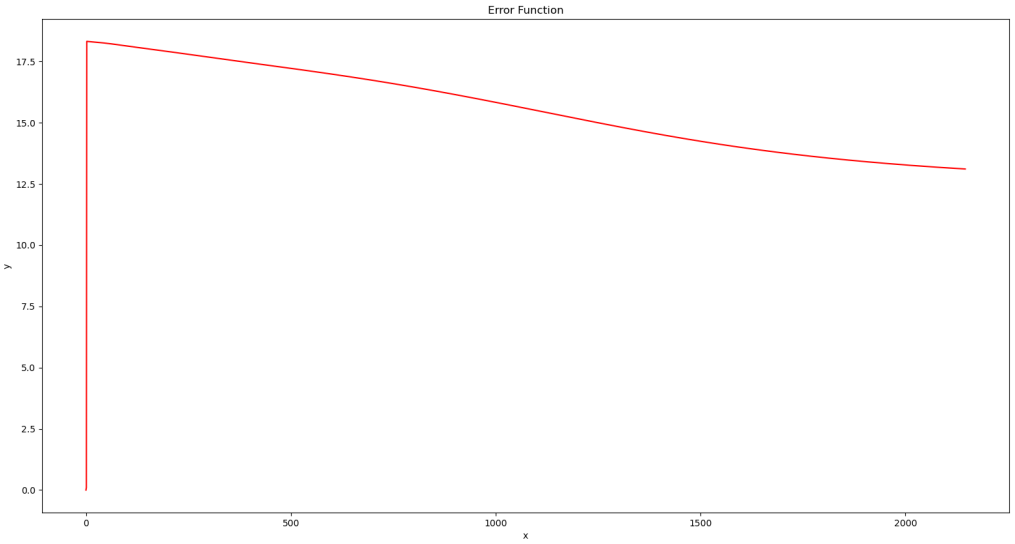
Final Output

```
$ python NonparametricRegressionTest.py
The RMSE for the kernel smoother with the Real_estate dataset is 13.109254100849096 with h = 2244
The squared errors for the kernel smoother with the Real_estate dataset is 171.85254308062883

Selected features for Real Estate data set:
1) transaction date
2) house age
3) distance to the nearest MRT station

The RMSE for the kernel smoother with the Insurance dataset is 12218.156011607187 with h = 226
The squared errors for the kernel smoother with the Insurance dataset is 149283336.32397282

Selected features for Insurance data set:
1) age
2) bmi
3) children
4) smoker
5) region

The RMSE for the kernel smoother with the CarPrice dataset is 7205.830341947259 with h = 1032
The squared errors for the kernel smoother with the CarPrice dataset is 51923990.91692774

Selected features in Car Price data set:
1) car height
2) compression ratio
3) horsepower
```

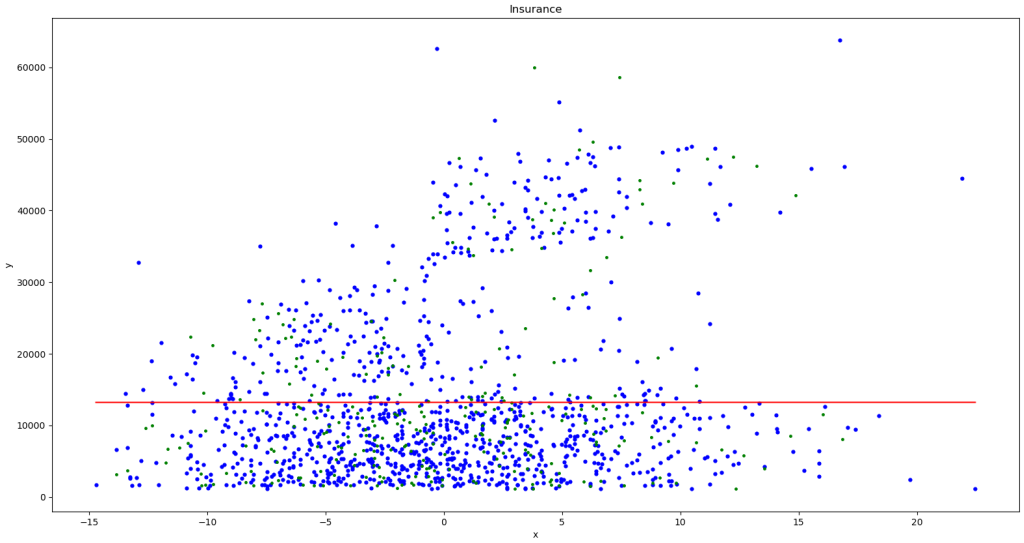
Nonparametric regression with the real estate data set



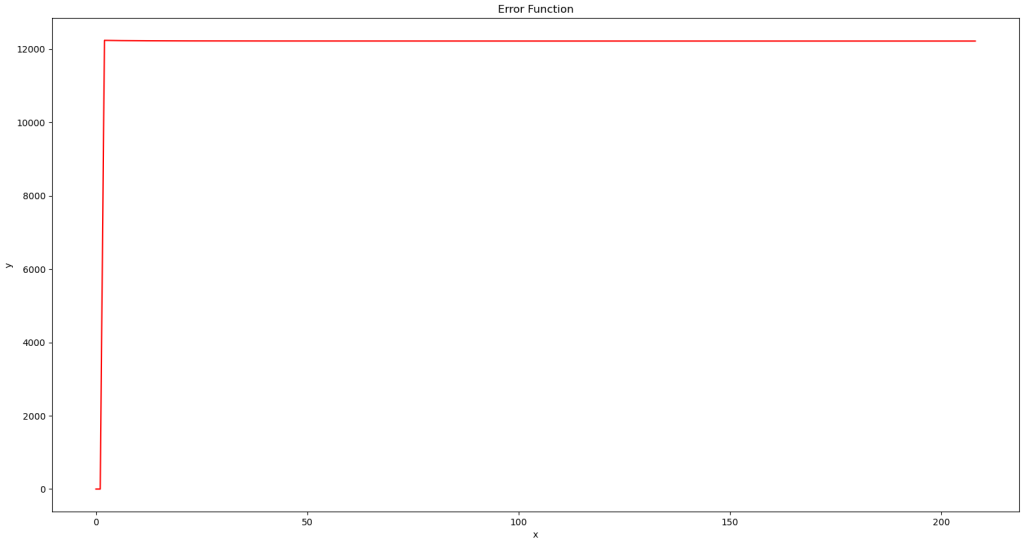
Optimizing *h* for the real estate data set



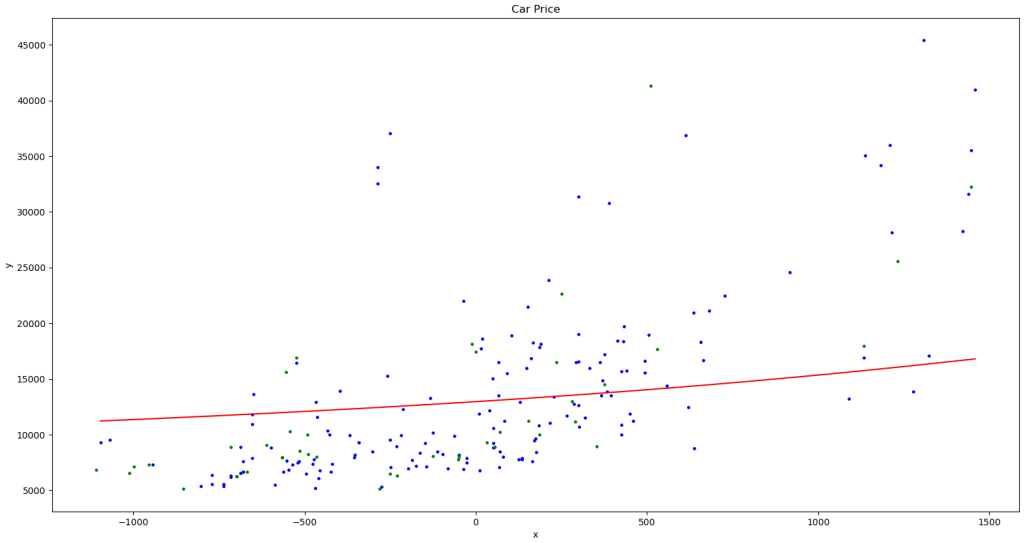
Nonparametric regression with the insurance data set



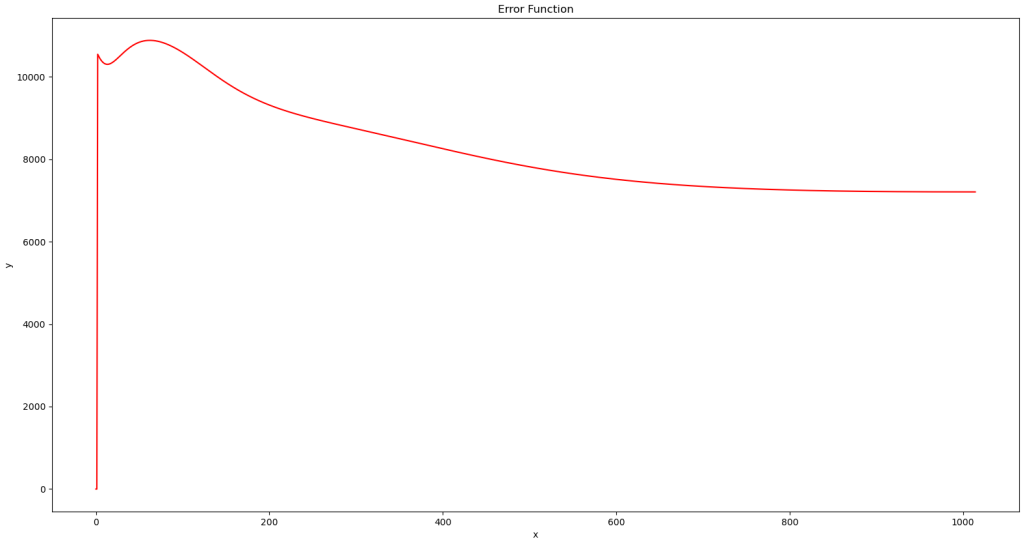
Optimizing *h* for the insurance data set



Nonparametric regression with the car price data set



Optimizing *h* for the car price data set

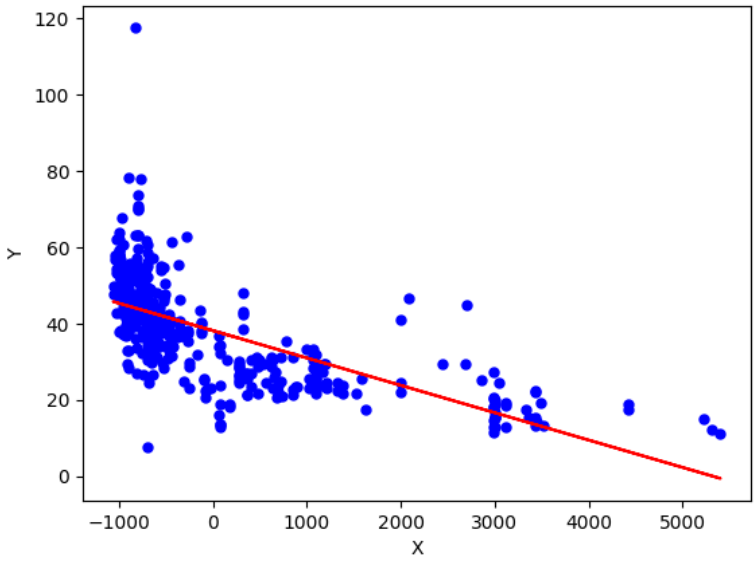


Linear Regression:

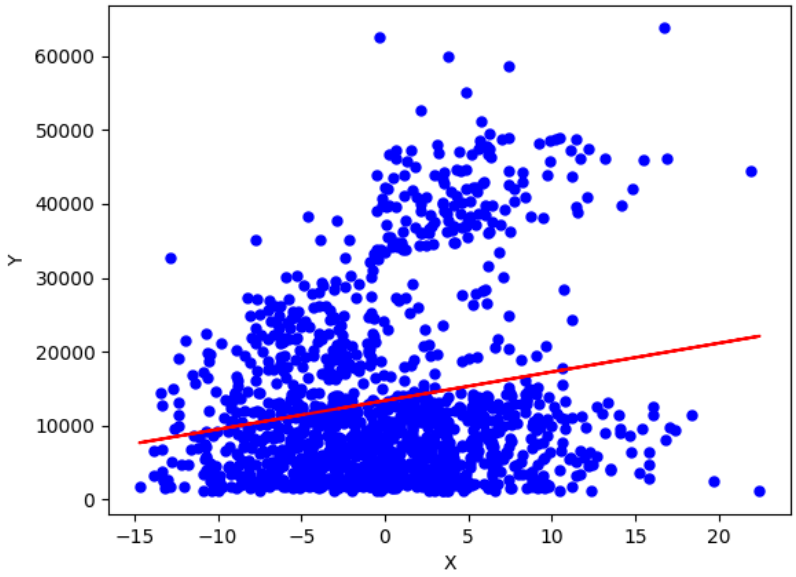
Final Output

```
Mouafak@LAPTOP-9F8615VP MINGW64 /d/University-materials/Comp-courses/QMBU 450/Regression Analysis/Regression-Analysis/Python format (main)
$ python RegressionTest.py
Warning: QT_DEVICE_PIXEL_RATIO is deprecated. Instead use:
   QT_AUTO_SCREEN_SCALE_FACTOR to enable platform plugin controlled per-screen factors.
   QT_SCREEN_SCALE_FACTORS to set per-screen factors.
   QT_SCALE_FACTOR to set the application global scale factor.

The RMSE for Linear Regression with the Real_estate.csv Dataset is  9.02717468366313
The MSE for Linear Regression with the Real_estate.csv Dataset is  81.48988276936853

Selected features for Real_estate.csv Dataset:
1) transaction date
2) house age
3) distance to the nearest MRT station

The RMSE for Linear Regression with the insurance.csv Dataset is  11345.744247245524
The MSE for Linear Regression with the insurance.csv Dataset is  128725912.52390492

Selected features for insurance.csv Dataset:
1) age
2) bmi
3) children

The RMSE for Linear Regression with the CarPrice_Assignment.csv Dataset is  5390.597784247966
The MSE for Linear Regression with the CarPrice_Assignment.csv Dataset is  29058544.47153908

Selected features for CarPrice_Assignment.csv Dataset:
1) car height
2) compression ratio
3) horsepower
```

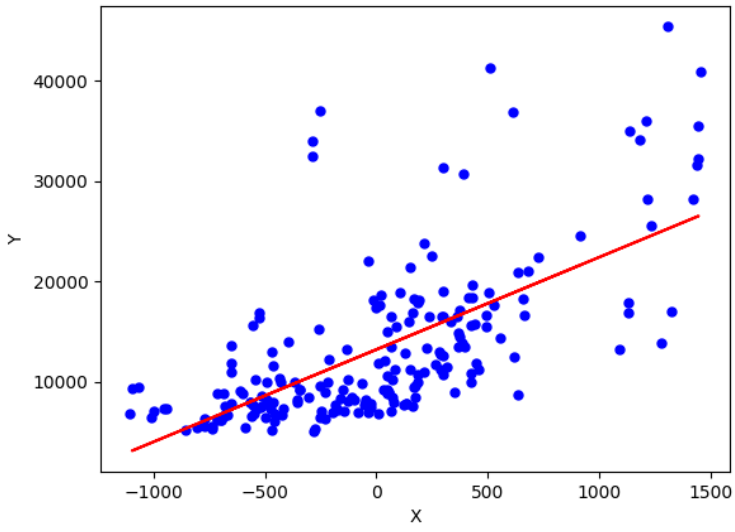
Linear Regression with The Real Estate Dataset



Linear Regression with The Insurance Dataset



Linear Regression with The Car-Price Dataset

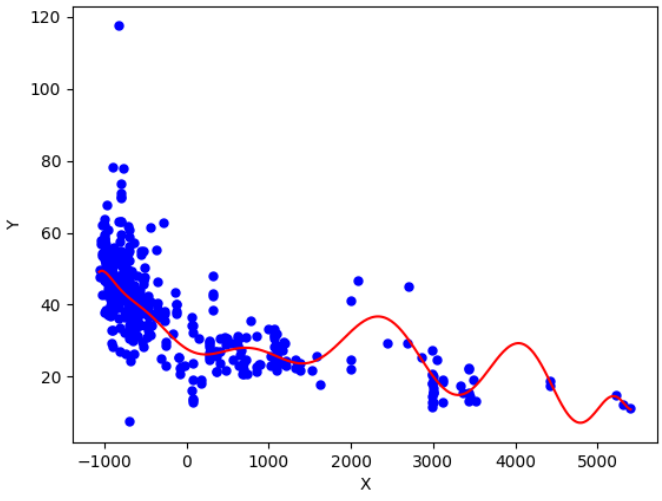


Polynomial Regression:

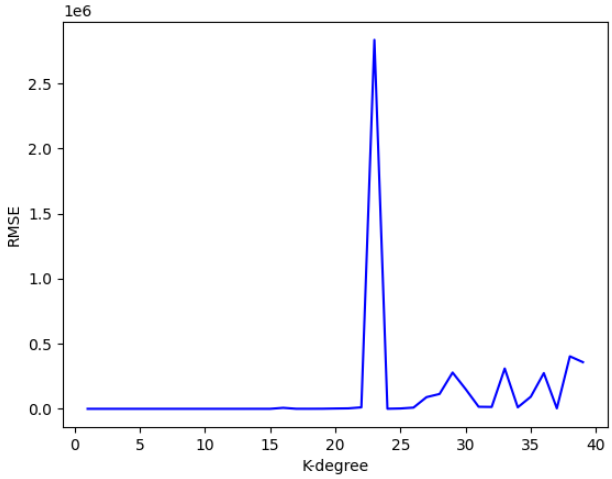
Final output

```
Mouafak@LAPTOP-9F8615VP MINGW64 /d/University-materials/Comp-courses/QMBU 450/Regression Analysis/Regression-Analysis/Python format (main)
$ python PolynomialRegressionTest.py
Warning: QT_DEVICE_PIXEL_RATIO is deprecated. Instead use:
   QT_AUTO_SCREEN_SCALE_FACTOR to enable platform plugin controlled per-screen factors.
   QT_SCREEN_SCALE_FACTORS to set per-screen factors.
   QT_SCALE_FACTOR to set the application global scale factor.

The RMSE for Polynomial Regression with the Real_estate.csv Dataset is  7.619366571357525  with K =  13
The MSE for Polynomial Regression with the Real_estate.csv Dataset is  58.054746948720535  with K =  13

Selected features for Real_estate.csv Dataset:

1) transaction date
2) house age
3) distance to the nearest MRT station
4) number of convenience stores
5) latitude

The RMSE for Polynomial Regression with the insurance.csv Dataset is  11330.057570819043  with K =  10
The MSE for Polynomial Regression with the insurance.csv Dataset is  128370204.55807391  with K =  10

Selected features for insurance.csv Dataset:
1) age
2) bmi
3) children

The RMSE for Polynomial Regression with the CarPrice_Assignment.csv Dataset is  4960.448248652905  with K =  4
The MSE for Polynomial Regression with the CarPrice_Assignment.csv Dataset is  24606046.82756367  with K =  4

Selected features for CarPrice_Assignment.csv Dataset:
1) car height
2) curb weight
3) compression ratio
4) horsepower
5) peak rpm
```

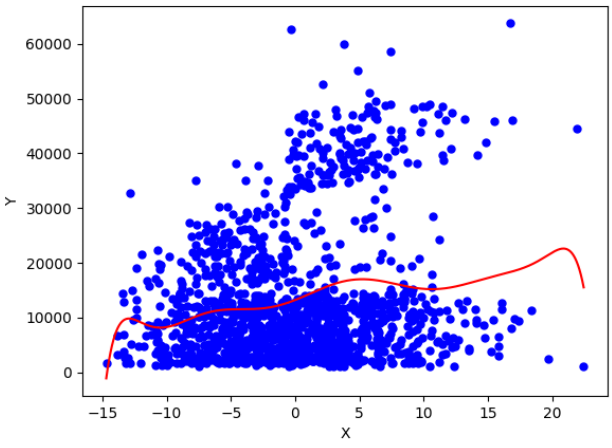
Polynomial Regression with The Real Estate Dataset



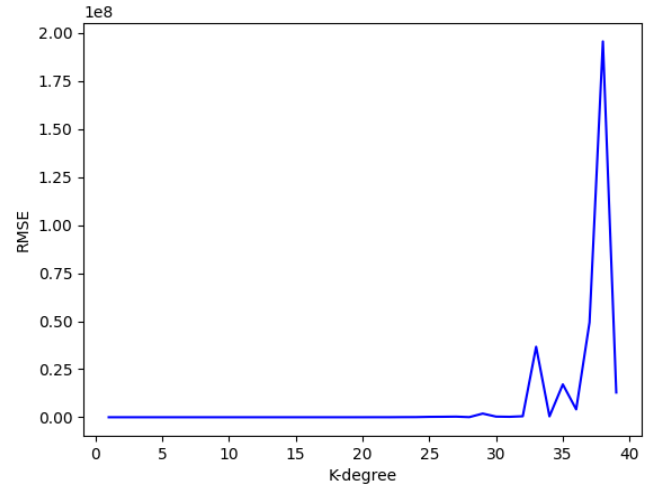
Optimization of K for The Real Estate Dataset



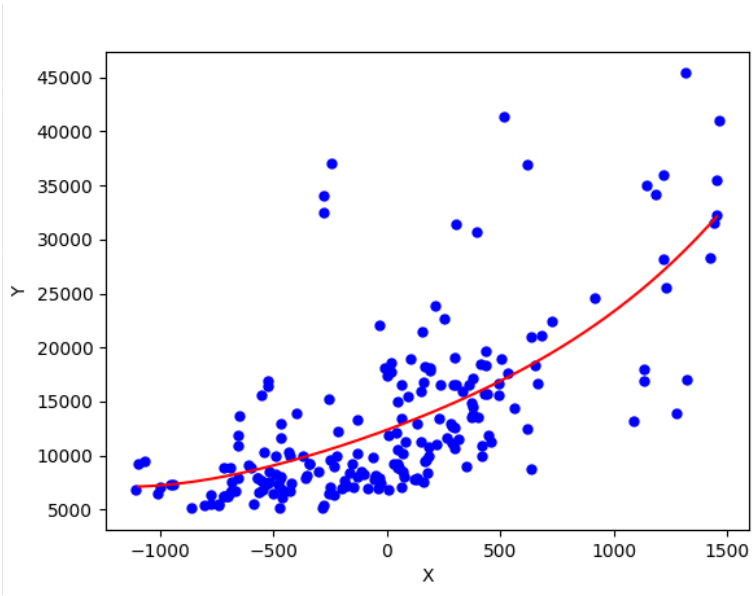
Polynomial Regression with The Insurance Dataset



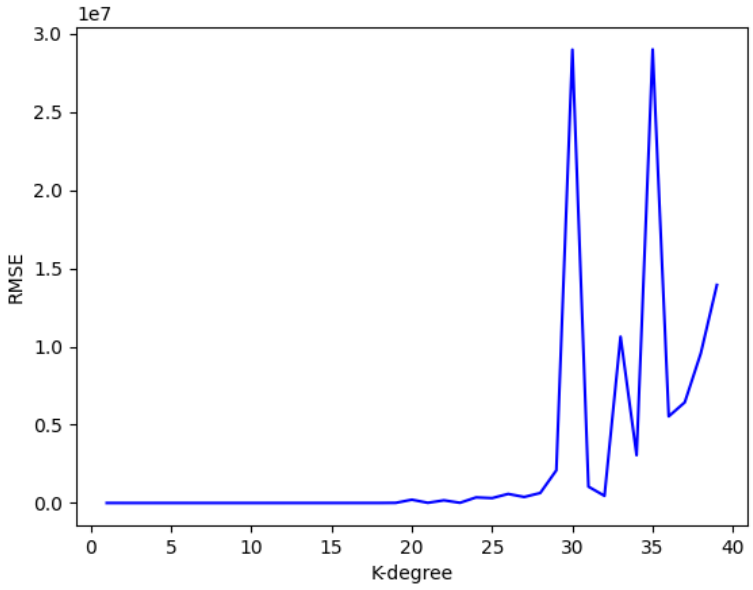
Optimization of K for The Insurance Dataset



Polynomial Regression with The Car-Price Dataset



Optimization of K for The Car-Price Dataset



References:

- Ethem Alpaydin - Introduction to Machine Learning (2014, The MIT Press)