CMPT 419 A1

Yu Ke
301414915

1. Probability

1. (a) $A = \{\text{" Infected with Malaria"}\}$
$B = \{\text{" test positive"}\}$

$P(A) = 0.01$

$P(B|A) = \dfrac{P(AB)}{P(A)} = 0.95 \quad P(B|\bar{A}) = \dfrac{P(\bar{A}B)}{P(\bar{A})} = 0.05$

$\therefore P(B) = P(AB) + P(\bar{A}B) = 0.95 \times 0.01 + 0.05 \times 0.99 = 0.059$

(b) $P(A|B) = \dfrac{P(AB)}{P(B)} = \dfrac{P(B|A) \times P(A)}{P(B)}$

$= 0.161$

2. $A = \{\text{rain today}\}$ $B = \{\text{rain tomorrow}\}$

$P(A) = 0.3 \quad P(B) = 0.6 \quad P(AB) = 0.25$

$P(B|A) = \dfrac{P(AB)}{P(A)} = 0.83$

3.

$P(\text{win}) = 0.1 + 0.2 + 0 = 0.3$

No, not fair

2. Weighted Squared Error

$$E_{\tilde{D}}(w) = \frac{1}{2}\sum_{n=1}^{N} a_n \{ t_n - w^T \Phi(x_n) \}^2$$

$$\frac{\partial}{\partial w} E_{\tilde{D}}(w) = \sum_{n=1}^{N} a_n \{ t_n - w^T \Phi(x_n) \} \cdot (-\Phi(x_n))$$

$$\nabla E_{\tilde{D}}(w) = \sum_{n=1}^{N} (a_n t_n - a_n w^T \Phi(x_n)) \cdot (-\Phi(x_n))^T$$

Set gradient to 0

$$0^T = \sum_{n=1}^{N} -a_n t_n \Phi(x_n)^T + \sum_{n=1}^{N} a_n w^T \Phi(x_n) \cdot \Phi(x_n)^T$$

$$\Phi = \begin{pmatrix} \Phi_0(x_1) & \cdots & \Phi_{M-1}(x_1) \\ \vdots & & \vdots \\ \Phi_0(x_N) & \cdots & \Phi_{M-1}(x_N) \end{pmatrix}$$

ab^T

- $\sum -a_n t_n \Phi(x_n)^T$

$= -(a_n t_n)^T \cdot \Phi \qquad = -t^T a^T \cdot \Phi$

- $\sum a_n w^T \left(\Phi(x_n) \Phi(x_n)^T \right)$

$= w^T \cdot a_n \cdot [\Phi_1(x_1) \cdots \Phi_{M-1}(x_N)] \begin{bmatrix} \Phi_1(x_1) \\ \vdots \\ \Phi_{M-1}(N) \end{bmatrix}$

$= w^T \cdot \Phi^T \cdot \alpha^T \cdot \Phi$

$\Rightarrow w = (\Phi^T a \Phi)^{-1} \cdot \Phi^T a t$

3. Training & Error

1. No. Because both training error and validation error are from sets which are randomly distributed. Training error will always underestimate validation error, but not less.

2. Yes. Because degree-10 contains degree-9, and degree-10 will fits better than degree-9 in unregularied regression.

3. No. Regularization reduces variance of the model, but it comes with a cost of increased bias in model.

   For the tradeoff, you can not garuanteed that regularization must be better.

4 Regression

4.1 Getting started

1.

```
import numpy as np
import pandas as pd
from scipy import nanmean
fname = 'SOWC_combined_simple.csv'
data = pd.read_csv(fname, na_values='_', encoding='latin1')
print(data[data['Under-5 mortality rate (U5MR) 1990'] == data['Under-5 mortality rate (U5MR) 1990'].min()])
```

```
   Countries and areas  Under-5 mortality rate (U5MR) 1990  \
76             Iceland                                 6.3
```

Iceland. The rate is 0.63%.

2.

```
print(data[data['Under-5 mortality rate (U5MR) 2011'] == data['Under-5 mortality rate (U5MR) 2011'].min()])
```

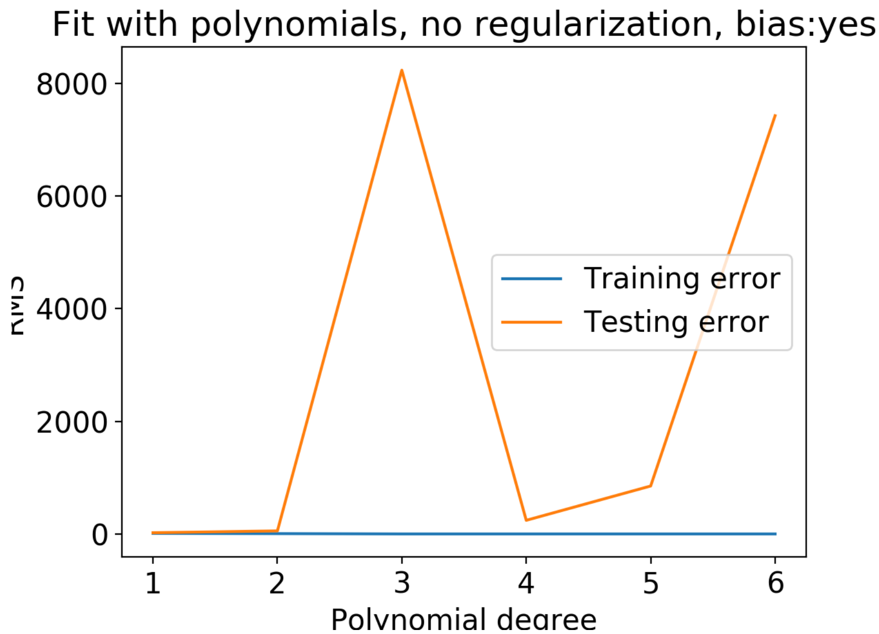
```
    Countries and areas  Under-5 mortality rate (U5MR) 1990  \
147          San Marino                                11.9

     Under-5 mortality rate (U5MR) 2011  U5MR male 2011  U5MR female 2011  \
147                                 1.8             1.8               1.8
```

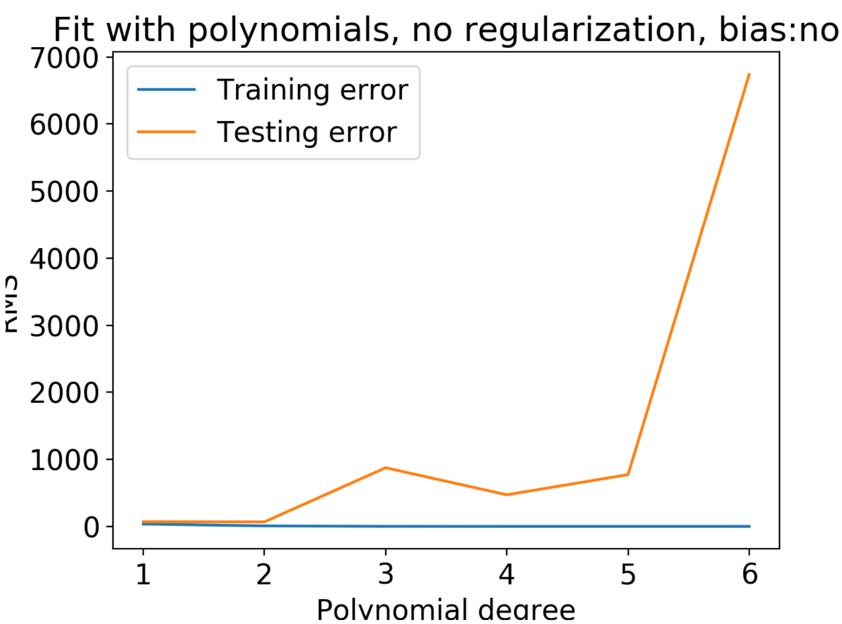
San Marino. The rate is 0.18%.

3. *na_values='_'* puts those values which is '_'to NAN. Then separate the values from features and countries and convert them to a numpy matrix. Next get the mean values of each column and give them to NAN values.

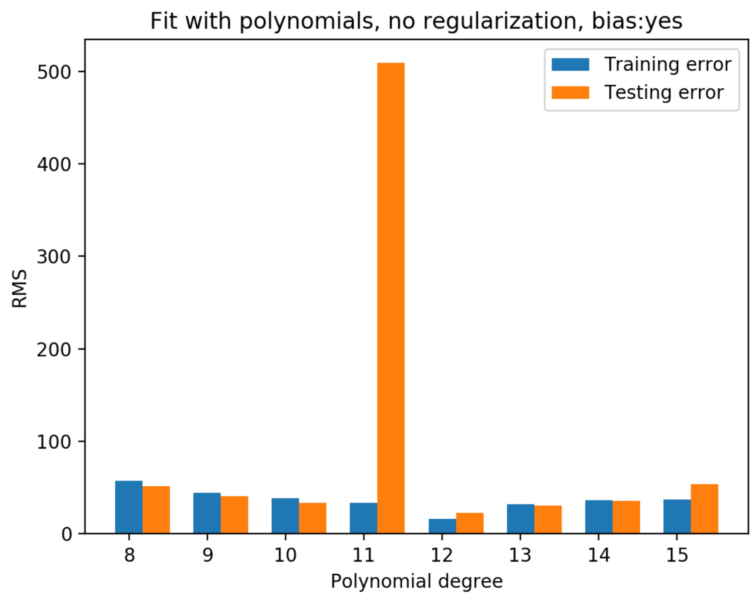
4.2 Polynomial Regression

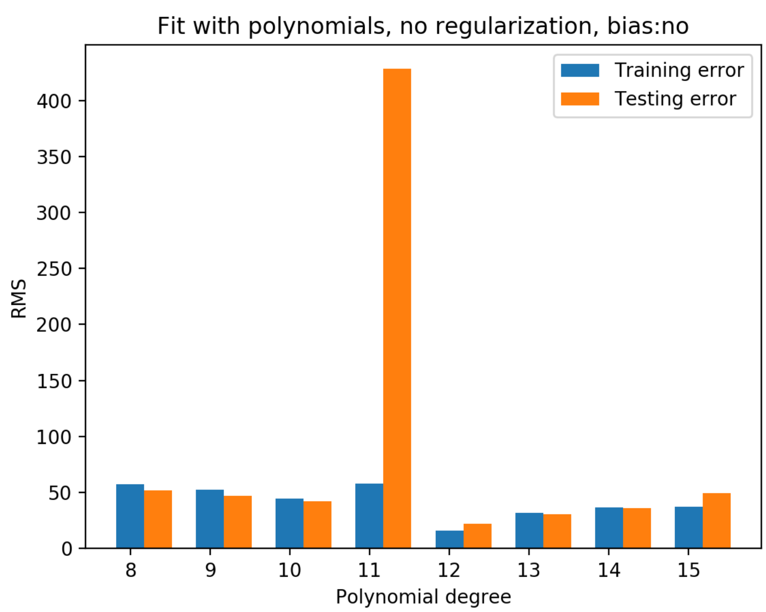
1.


Fit with polynomials, no regularization, bias:yes
8000
6000
4000
2000
0
RMS
Training error
Testing error
1
2
3
4
5
6
Polynomial degree

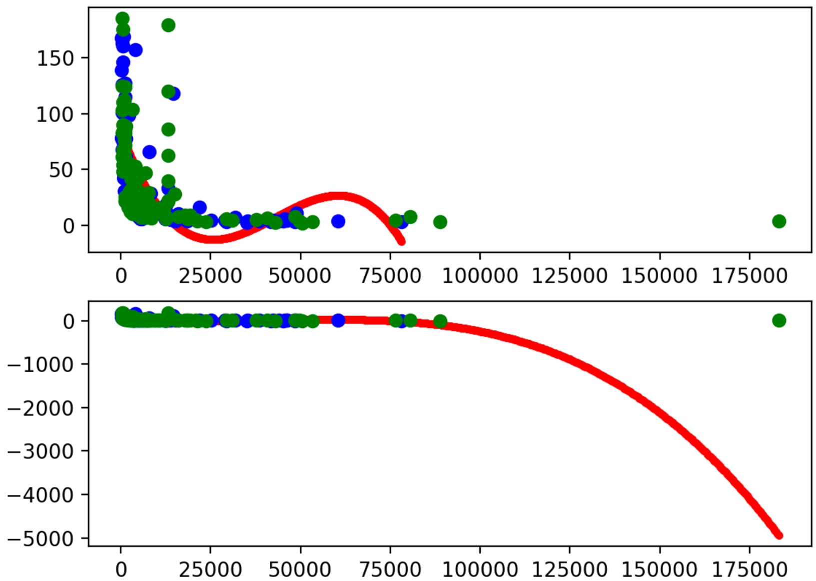

Fit with polynomials, no regularization, bias:no
7000
6000
5000
4000
3000
2000
1000
0
RMS
Training error
Testing error
1
2
3
4
5
6
Polynomial degree

2.

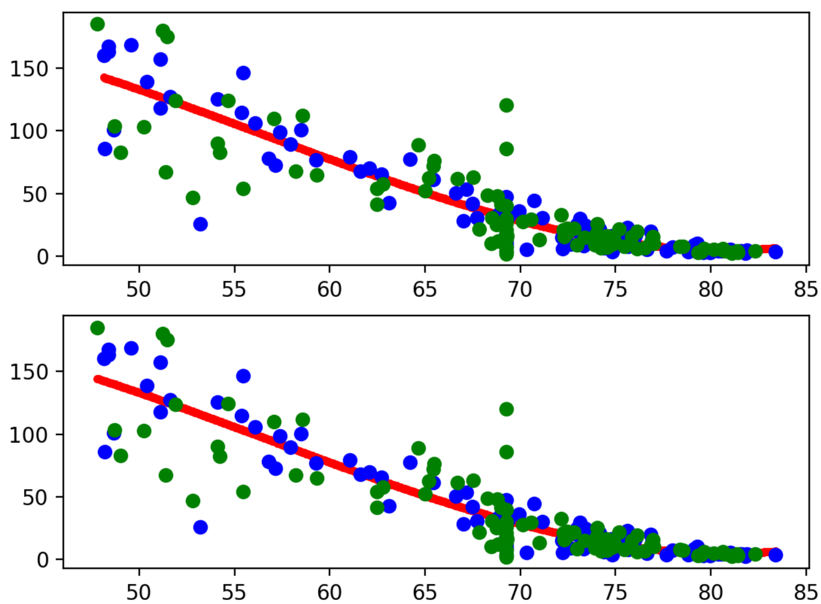

Fit with polynomials, no regularization, bias:yes
Training error
Testing error
RMS
0
100
200
300
400
500
8
9
10
11
12
13
14
15
Polynomial degree


Fit with polynomials, no regularization, bias:no
Training error
Testing error
RMS
0
50
100
150
200
250
300
350
400
8
9
10
11
12
13
14
15
Polynomial degree

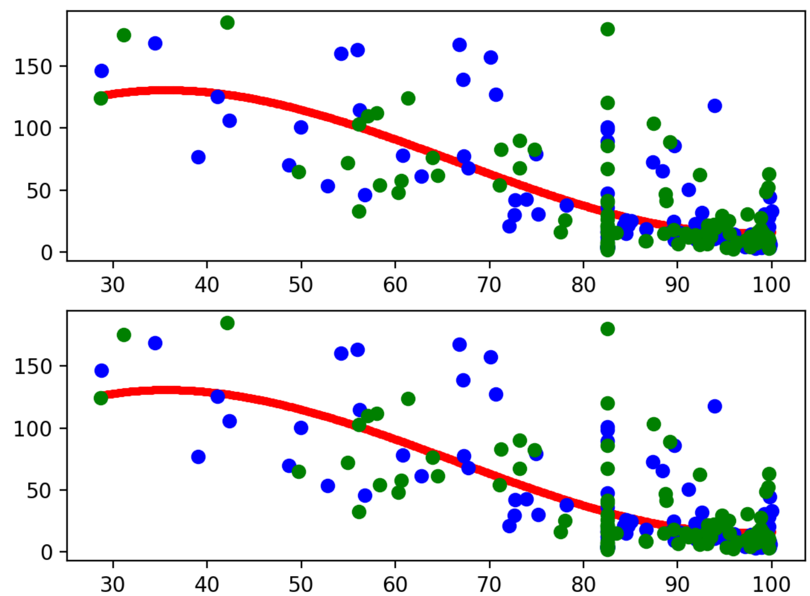
Visulization of feature 11



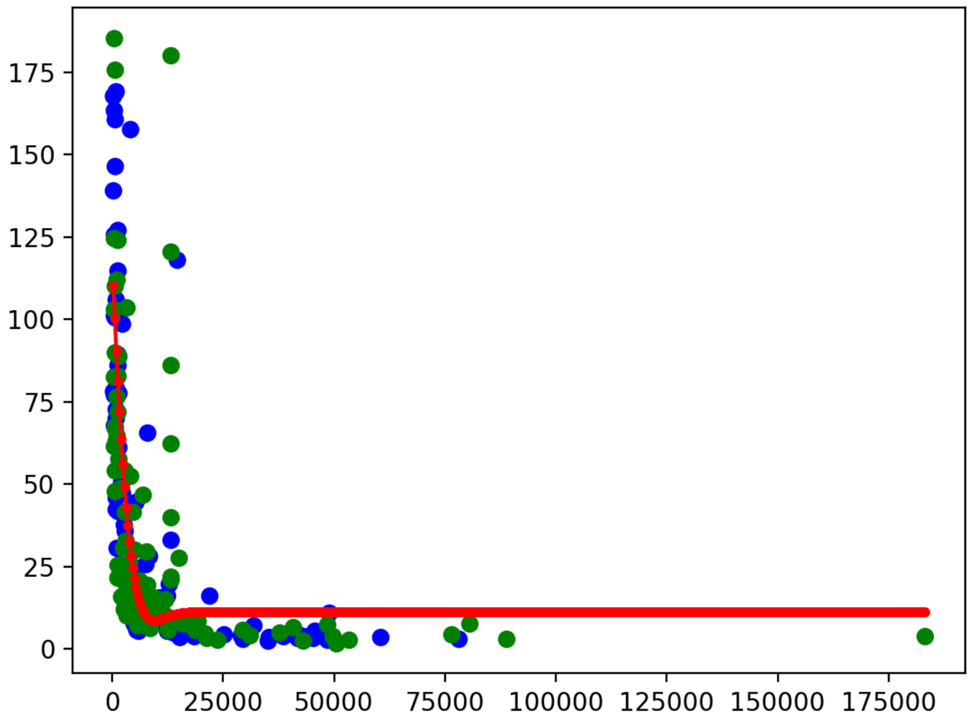
Visulization of feature 12



Visulization of feature 13



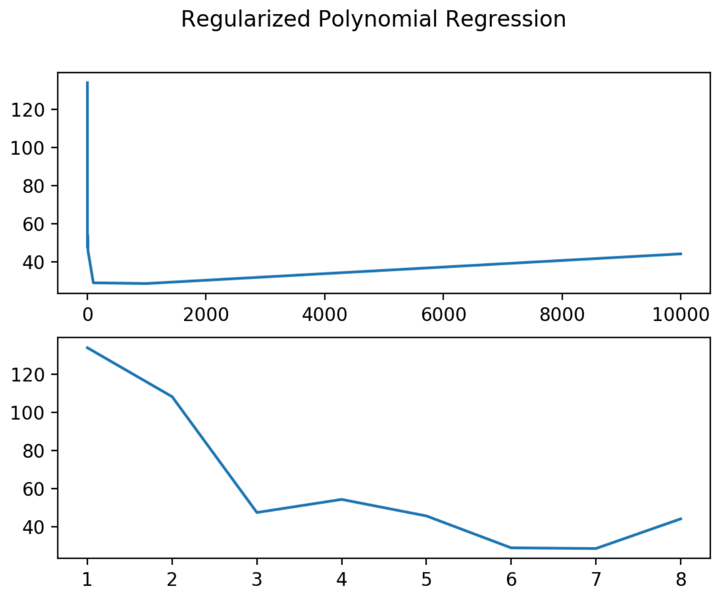
4.3 Sigmoid Basis Functions



train error = 28.457937762731884

test error = 33.806724900990964

4.4 Regularized Polynomial Regression



The x axis of the plot in the top is [0,.01,.1,1,10,100 ,1000 ,10000]
The x axis of the plot in the bottom is the sequence.

Average validation set error :

λ	Average validation set error
0	134.0872480012021
0.01	108.33938137985342
0.1	47.42014676706832
1	54.301530186869215,
10	45.617478912705465
100	28.827211650792673
1000	28.469223279788757
10000	44.06076704806574