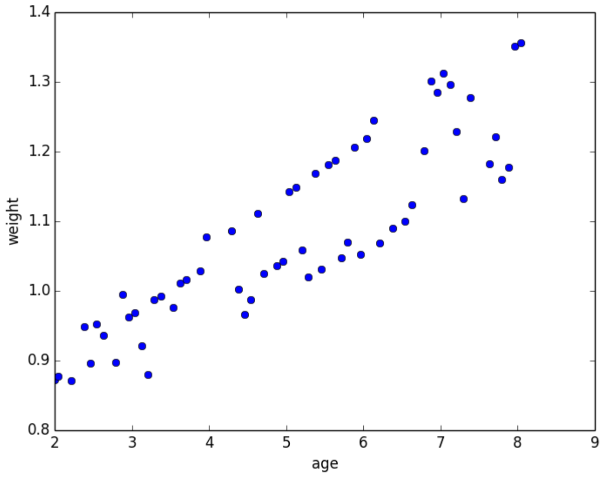
Problem 1

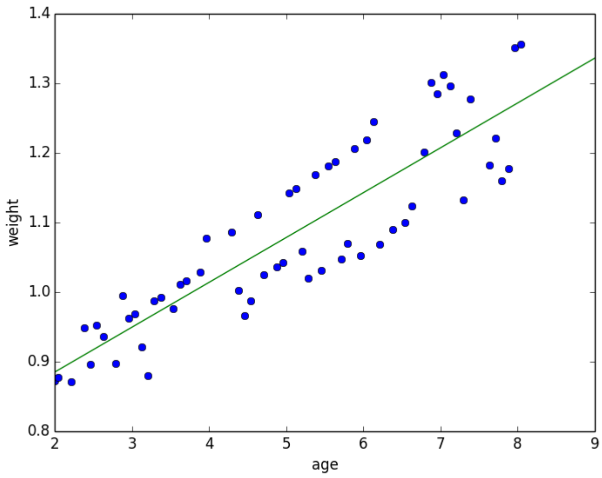
1.1 Plot the distribution of the data.



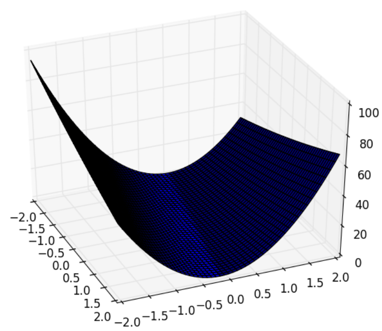
1.2 The mean square error of the model on the training set?

Answer: 0.00364981533271

1.3 (a) Plot the regression line on top of the distribution.



1.3 (b) Plot the cost function.



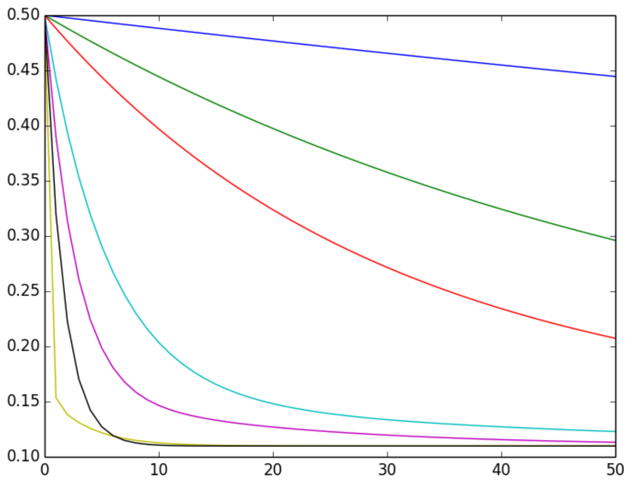
1.4 (a) Make a prediction for a 4.5 years old girl.

Answer: 1.04616421223

1.4 (b) Compute the mean square error of the test set.

Answer: 0.00640555075731. As we can see, it is very large compared to the training set.

2.3 (a) Plot the cost function for different α.



2.3 (b) Compare the convergence rate for different α.

Answer: larger alpha will result in quick convergence

2.3 (c) What is the best α and the corresponding β?

Answer: the best α is 1,

$\beta_0 =$ 8.4742339727717663e-16

$\beta_1 =$ 0.87640906401391461

$\beta_2 =$ 0.0095566478680290702

2.3 (d) Height prediction for a 5-year old girl weighting 20 kilos.

Answer: 1.08259528743

2.4 (a) Compare the β obtained with gradient descendant.

Answer: the β obtained by normal equation is shown below, they are different because we scale the data when performing gradient descendant

$\beta_0 =$ 7.39451312e-01

$\beta_1 =$ 6.77117569e-02

$\beta_2 =$ 2.29259412e-04

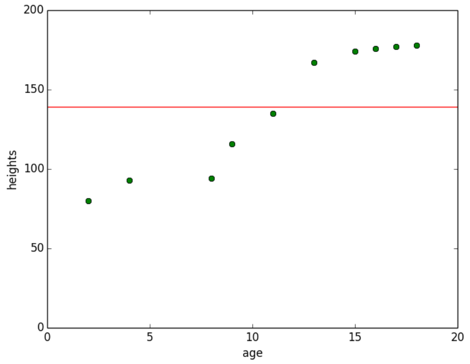
2.4 (b) Make a height prediction for a 5-year old girl weighting 20 kilos

Answer: 1.08259528491. Compared to gradient descendant, the results are very similar but not the same.

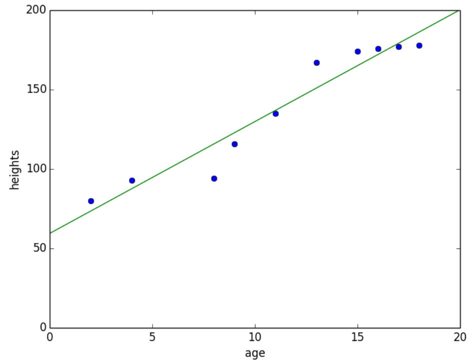
(optional part)

3.2 Plot the data points obtained for each degree

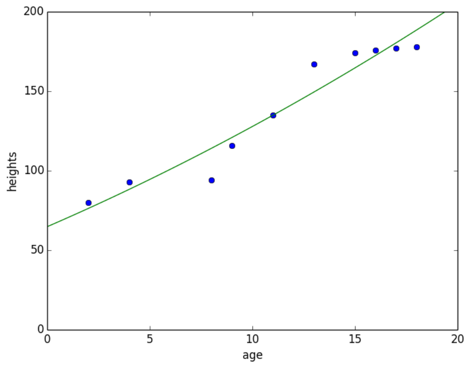
d = 0



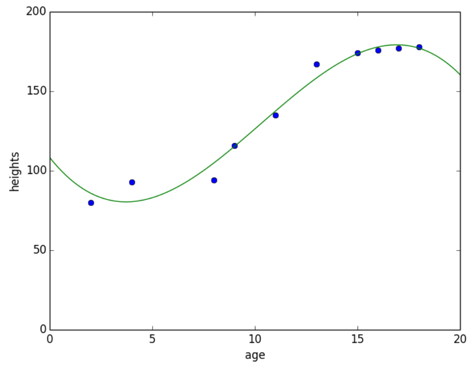
d = 1



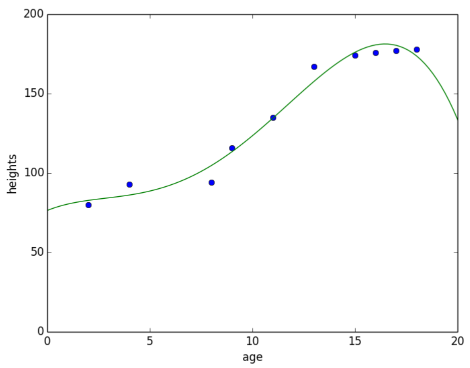
d = 2



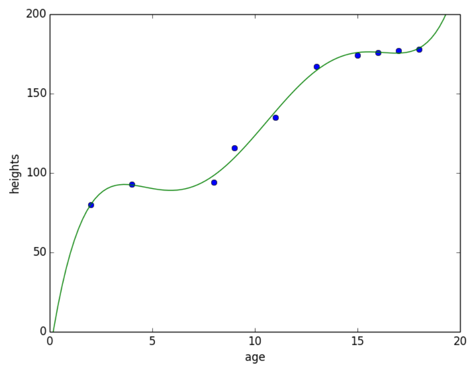
d = 3



d = 4



d = 5



3.3 For what degree do you observe over-fitting or under-fitting?

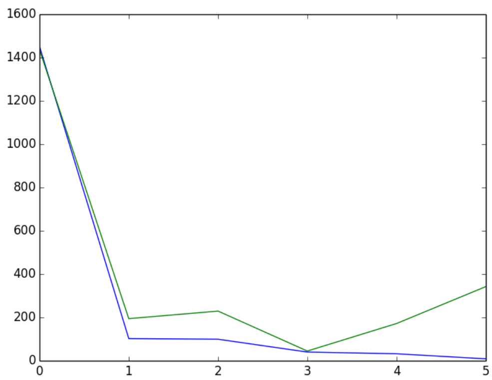
Answer: apparently it is under-fitting for d = 0, we use a constant to estimate all girls height. For d = 5, we have over-fitting because the curve turns very quickly toward any data sample

3.4 Use validation data to pick the best d.

Answer: the mean square errors are listed below

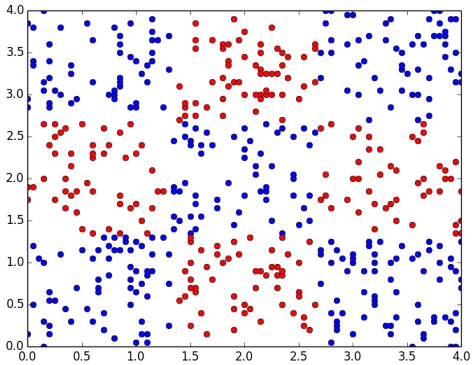
	d = 0	d = 1	d= 2	d= 3	d = 4	d= 5
training	1451	102	99	40	32	8.7
validation	1436	194	229	44	171	342

Below is the plot for the errors. Green line plots the errors of validation set, and blue line plots the errors of training set. From the plot we can see d = 3 is the best. It has the lowest validation and training error.



Problem 2

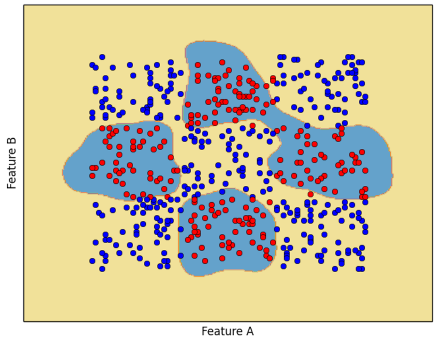
2. Plot a scatter plot for the data.



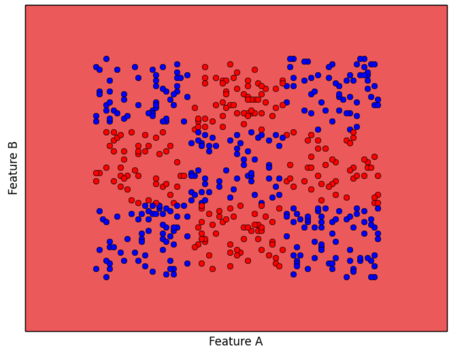
3 & 4. Plot the results and compare different C and polynomial degree.

Answer: I set parameter C = 1 and polynomial degree = 6 and gamma = 10 to perform SVM. I have performed SVM, logistic regression, perceptron and ensemble method. The results are shown below. I used 3-fold cross validation.

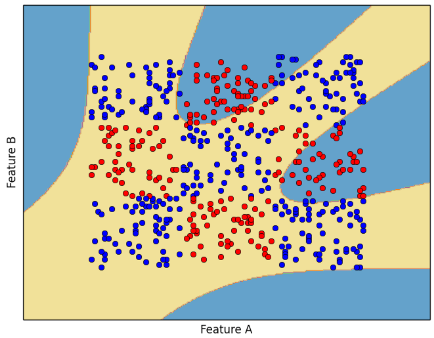
SVM-RBF



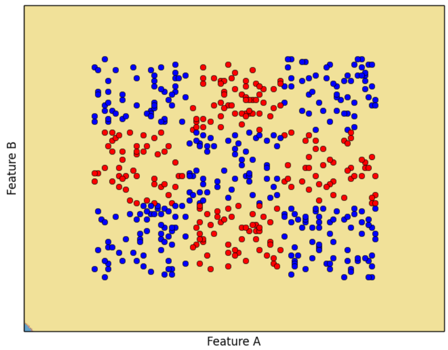
SVM-Linear



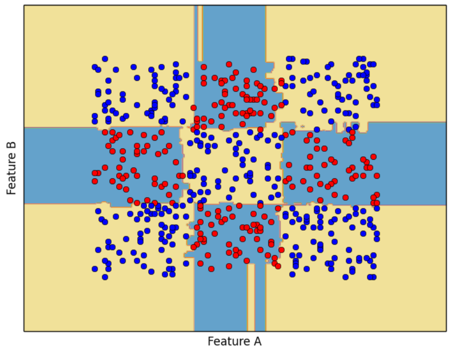
SVM-Polynomial



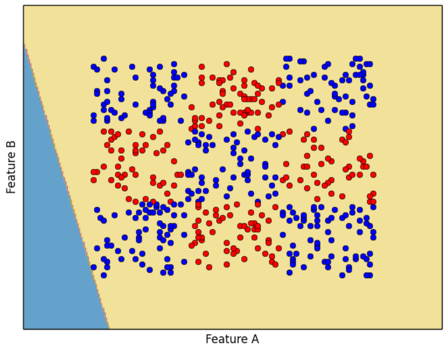
Logistic Regression



Random forest classifier



Perceptron



As we can see from the plots, logistic regression, perceptron, and linear SVM have nearly no effect on chessboard shape data. I think it is because they can only produce linear boundary, hence have no good solution for the chessboard-shaped data. On the other hand, random forest classifier and SVM-RBF perform very well. Although SVM-polynomial is providing relevant prediction here, it has a lot of errors.

Below is a table to compare the best and worst score for each method.

	SVM RBF	SVM Linear	SVM Polynomial	Logistic regression	Random forest	Perceptron
Score	0.96	0.59	0.72	0.59	0.96	0.59

Observe the data give us the following result:
For SVM-RBF, increase gamma will have a better result, but increase too much will not have much benefit. For SVM-Polynomial, increase the degree will give us a better result, but with degree grater than 6, the result gets worse. Changing C will not have too much impact on the result.