RICE UNIVERSITY, DEPARTMENT OF COMPUTER SCIENCE

Assignment 1, COMP 540

Chen Zeng(cz39), Zhihui Xie(zx18)

January 15, 2018

1 Background refresher

Solution 1

- Plot the histogram of samples generated by a categorical distribution with probabilities [0.2,0.4,0.3,0.1].
  Answer: Figure 1.1 shows the result.
- Plot the univariate normal distribution with mean of 10 and standard deviation of 1.
  Answer: Figure 1.2 shows the result.
- Produce a scatter plot of the samples for a 2-D Gaussian with mean at [1,1] and a covariance matrix [[1,0.5],[0.5,1]].
  Answer: Figure 1.3 shows the result.
- Test your mixture sampling code by writing a function that implements an equal-weighted mixture of four Gaussians in 2 dimensions, centered at $(\pm 1, \pm 1)$ and having covariance I. Estimate the probability that a sample from this distribution lies within the unit circle centered at (0.1, 0.2) and include that number in your writeup.
  Answer: The estimated probability is 0.18075.

Solution 2 Here we suppose that there are two independent Poisson Random Variables and we define their moment-generating function as the following:

$$X \sim P(\lambda_1), Y \sim P(\lambda_2)$$

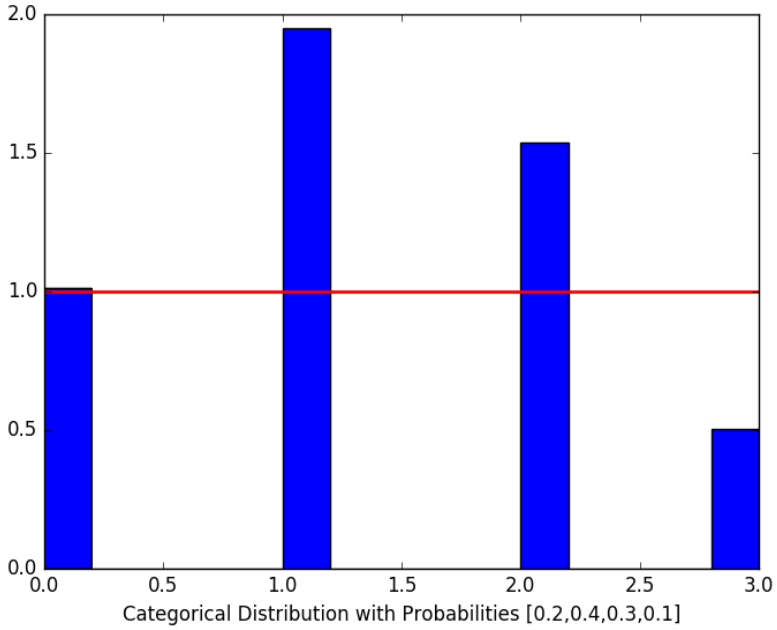


Figure 1.1: Categorical Distribution

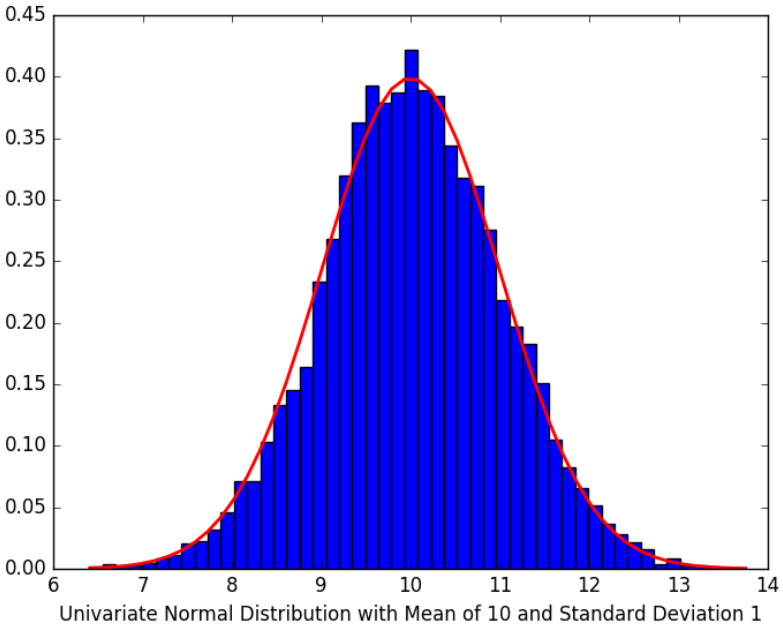


Figure 1.2: Univariate Normal Distribution

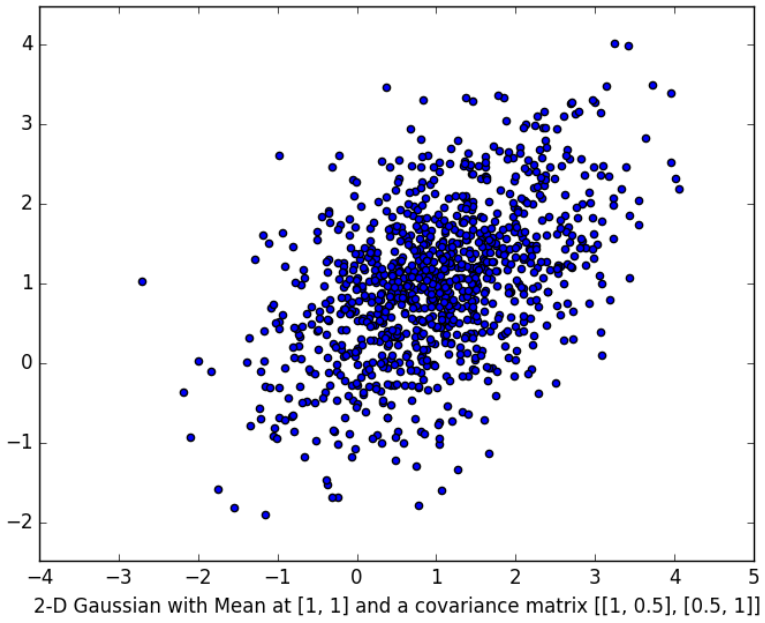


Figure 1.3: MultiVariate Normal Distribution

According to the properties of Poisson Random Variables, we can get:

$$P(X=x)=\frac{\lambda_1^x}{x!}e^{-\lambda_1}, P\left(Y=y\right)=\frac{\lambda_2^y}{y!}e^{-\lambda_2}$$

Therefore, using the binomial formula and some transformation, we can get:

$$\begin{aligned} P(X+Y=z) &= \sum_{x=0}^{z}\frac{\lambda_1^x}{x!}e^{-\lambda_1}\frac{\lambda_2^{z-x}}{(z-x)!}e^{-\lambda_2} \\ &= e^{-(\lambda_1+\lambda_2)}\sum_{x=0}^{z}\frac{\lambda_1^x}{x!}\frac{\lambda_2^{z-x}}{(z-x)!} \\ &= e^{-(\lambda_1+\lambda_2)}\frac{(\lambda_1+\lambda_2)^z}{z!} \end{aligned}$$

According to the definition, $X+Y$ is also a Poisson random variable

Solution 3 According to transformation, we can get:

$$\begin{aligned} p(X_1 = x_1) &= \int p(X_1 = x_1, X_0 = x_0)\, dx_0 \\ &= \int p(X_0 = x_0)\, p(X_1 = x_1 | X_0 = x_0)\, dx_0 \\ &= \alpha_0 \alpha \int exp\left(-\frac{1}{2}\left(\frac{(x_0 - \mu_0)^2}{\sigma_0^2} + \frac{(x_1 - x_0)^2}{\sigma^2}\right)\right) dx_0 \\ &= \frac{\alpha_0 \alpha}{A} exp\left(-\frac{1}{2}\frac{(x_1 - \mu_0)^2}{\sigma^2 + \sigma_0^2}\right) \end{aligned}$$

Here A is a constant. At the same time, according to the given condition we can know:

$$p(X_1 = x_1) = \alpha_1 exp\left(-\frac{1}{2}\frac{(x_1 - \mu_1)^2}{\sigma_1^2}\right)$$

Therefore, we can get the relations:

$$\mu_1 = \mu_0$$

$$\sigma_1^2 = \sigma^2 + \sigma_0^2$$

$$\alpha_1 = \frac{\alpha_0 \alpha}{A} = \alpha_0 \alpha \int exp\left(-\frac{1}{2}\frac{x_0^2 - 2\left(\frac{\sigma^2 \mu_0 + \sigma_0^2 x_1}{\sigma^2 + \sigma_0^2}\right) x_0 + \left(\frac{\sigma^2 \mu_0 + \sigma_0^2 x_1}{\sigma^2 + \sigma_0^2}\right)^2}{\sigma_0^2 \sigma^2 / (\sigma^2 + \sigma_0^2)}\right) dx_0$$

Solution 4 As for eigenvalues, we just need to solve:

$$(13 - \lambda)(4 - \lambda) = 10$$

And we can get the result:

$$\lambda_1 = 3, \lambda_2 = 4$$

As for eigenvectors, there are two cases:

when $\lambda_1 = 3$: $A - \lambda I = \begin{bmatrix} 10 & 5 \\ 2 & 1 \end{bmatrix}$, and the according eigenvector is $\begin{bmatrix} 1 \\ -2 \end{bmatrix}$

when $\lambda_2 = 14$: $A - \lambda I = \begin{bmatrix} -1 & 5 \\ 2 & -10 \end{bmatrix}$, and the according eigenvector is $\begin{bmatrix} 5 \\ 1 \end{bmatrix}$

Solution 5 As for $(A + B)^2 \neq A^2 + 2AB + B^2$, we get the following example:

$$A = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & -1 \\ 0 & -1 \end{bmatrix}$$

As for $AB = 0, A \neq 0, B \neq 0$, we get the following example:

$$A = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

Solution 6

$$\begin{aligned} A^T A &= \left(I-2uu^T\right)^T \left(I-2uu^T\right) \\ &= \left(I-2uu^T\right)\left(I-2uu^T\right) \\ &= I-4uu^T+4uu^T uu^T \\ &= I-4uu^T+4u\left(u^T u\right)u^T \\ &= I-4uu^T+4uu^T = I \end{aligned}$$

Solution 7-1 $f(x)=x^3, f'(x)=3x^2, f''(x)=6x$
Here f'' should always be non-negative when $x \geqslant 0$
Therefore, $f(x)=x^3$ is convex for $x \geqslant 0$

Solution 7-2 According to the property of f, and the fact that $\lambda, 1-\lambda \in [0,1]$ we can get the following transformation:

$$\begin{aligned} & f\left(\lambda(x_1,x_2)+(1-\lambda)\left(y_1,y_2\right)\right) \\ &= f\left(\lambda x_1+(1-\lambda)y_1, \lambda x_2+(1-\lambda)y_2\right) \\ &= max\left(\lambda x_1+(1-\lambda)y_1, \lambda x_2+(1-\lambda)y_2\right) \\ &\leqslant \lambda max(x_1,x_2)+(1-\lambda)\,max\left(y_1,y_2\right) \\ &= \lambda f(x_1,x_2)+(1-\lambda)f\left(y_1,y_2\right) \end{aligned}$$

Therefore:

$$f\left(\lambda(x_1,x_2)+(1-\lambda)\left(y_1,y_2\right)\right) \leqslant \lambda f(x_1,x_2)+(1-\lambda)f\left(y_1,y_2\right)$$

It proves that f is convex

Solution 7-3 Let $h=f+g$, then as for $\lambda \in [0,1]$ and $x, y \in S$, we can get:

$$\begin{aligned} & h\left(\lambda x+(1-\lambda)y\right) \\ &= f\left(\lambda x+(1-\lambda)y\right)+g\left(\lambda x+(1-\lambda)y\right) \\ &\leqslant \lambda f(x)+(1-\lambda)f\left(y\right)+\lambda g(x)+(1-\lambda)g\left(y\right) \\ &= \lambda\left(f(x)+g(x)\right)+(1-\lambda)\left(f\left(y\right)+g\left(y\right)\right) \\ &= \lambda h(x)+(1-\lambda)h\left(y\right) \end{aligned}$$

Therefore:

$$h\left(\lambda x+(1-\lambda)y\right) \leqslant \lambda h(x)+(1-\lambda)h\left(y\right)$$

It proves that $h=f+g$ is convex

Solution 7-4 Let $h = fg$
Here are two facts:
1. f and g are convex, therefore: $f'', g'' \geqslant 0$
2. f and g are non-negative, therefore: $f, g \geqslant 0$
Because of these two facts, therefore: $f''g, fg'' \geqslant 0$
Also, because f and g are convex, non-negative, and have their minimum within S at the same point, therefore: $f'g' \geqslant 0$
Therefore:

$$h'' = (fg)'' = f''g + fg'' + 2f'g' \geqslant 0$$

It proves that $h = fg$ is convex

Solution 8 Accoding to the constraint, we can get:

$$L = -\sum_{i=1}^{K} p_i log p_i + \lambda \left(\sum_{i=1}^{K} p_i - 1 \right)$$

Then let's take the derivation of it and we can get:

$$\frac{\partial L}{\partial p_i} = -log p_i - 1 + \lambda = 0$$

And according to the constraint:

$$\sum_{i=1}^{K} p_i - 1 = 0$$

Therefore:

$$p_1 = p_2 = ... = p_K = \frac{1}{K}$$

2 Locally weighted linear regression

Solution 1 Expand the equation and compare it with the minimized function, we can get:

$$\vec{W} = \frac{1}{2} \begin{bmatrix} w^{(1)} & & & \\ & w^{(2)} & & \\ & & ... & \\ & & & w^{(m)} \end{bmatrix}$$

Solution 2 As for generalizing the normal equation to the weighted setting, which means $w^{(i)}$ are not equal to 1, let's still take partial of J by θ and let it be 0:

$$J(\theta) = (X\theta - y)^T W (X\theta - y)$$

$$\frac{\partial J}{\partial \theta} = \frac{\partial}{\partial \theta} (X\theta - y)^T W (X\theta - y) = 0$$

To make the partial calculation to be more clear, let's look back to the original J without vectors expression.

$$J(\theta) = \frac{1}{2}\sum_{i=1}^{m} w^{(i)}\left(\theta^T x^{(i)} - y^{(i)}\right)^2 = \frac{1}{2}\sum_{i=1}^{m} w^{(i)}\left[\left(\sum_{j=1}^{d} x_j^{(i)}\theta_j\right) - y^{(i)}\right]^2$$

Therefore:

$$\frac{\partial}{\partial\theta_j}J(\theta) = \sum_{i=1}^{m} w^{(i)}\left[\left(\sum_{j=1}^{d} x_j^{(i)}\theta_j\right) - y^{(i)}\right]x_j^{(i)}$$

Thus, as for matrix expression:

$$\frac{\partial}{\partial\theta}J(\theta) = X^T W\left(X\theta - y\right)$$

Let it be 0, we can get the following equation:

$$X^T W\left(X\theta - y\right) = 0$$

Thus

$$\theta = \left(X^T W X\right)^{-1} X^T W y$$

Solution 3 According to the algorithm of batch gradient descent:

$$\theta_j \leftarrow \theta_j - \alpha\sum_{i=1}^{m} w^{(i)}\left(\theta^T x^{(i)} - y^{(i)}\right)x_j^{(i)}$$

Locally weighted linear regression a non-parametric method

3 Properties of the linear regression estimator

Solution 1 According to the closed form solution and the given conditions:

$$\theta^* = \left(X^T X\right)^{-1} X^T y = \left(X^T X\right)^{-1} X^T\left(X\theta + \varepsilon\right) = \theta + \left(X^T X\right)^{-1} X^T \varepsilon$$

Then we can calculate it's expectation:

$$\begin{aligned} E[\theta] &= E\left[\theta^* - \left(X^T X\right)^{-1} X^T \varepsilon\right] \\ &= \theta^* - \left(X^T X\right)^{-1} X^T E[\varepsilon] \\ &= \theta^* \end{aligned}$$

Solution 2 According to the given conditions and some properties, we can get:

$$\begin{aligned}
Var[\theta] &= E\left[(\theta-\theta^*)(\theta-\theta^*)^T\right] \\
&= E\left[(\theta^*-\theta)(\theta^*-\theta)^T\right] \\
&= E\left[\left((X^TX)^{-1}X^T\varepsilon\right)\left((X^TX)^{-1}X^T\varepsilon\right)^T\right] \\
&= E\left[(X^TX)^{-1}X^T\varepsilon\varepsilon^TX(X^TX)^{-1}\right] \\
&= E\left[\varepsilon\varepsilon^T\right]E\left[(X^TX)^{-1}X^TX(X^TX)^{-1}\right] \\
&= \sigma^2(X^TX)^{-1}
\end{aligned}$$

4 Implement linear regression & regularized linear regression

Part 1.A

- **Question**: What can you say about the quality of the linear fit for this data? Explain how you expect the model to perform at the low and high ends of values for LSTAT? How could we improve the quality of the fit?
  Answer: The quality of the linear fit is great and we can see that it can basically fit the distribution of data in the plot. At the low end of values for LSTAT, the value of y should be larger. And at the high end of values for LSTAT, the value of y should be smaller. To improve the quality of the fit, maybe the loss function can be modified and more data are needed to train the model.

- **Question**: Fill in code for prediction using the computed θ at the indicated point in the box below. Report the predictions of your model in your writeup.pdf.
  Answer:
  - For lower status percentage = 5, we predict a median home value of [298034.49412207];
  - For lower status percentage = 50, we predict a median home value of [-129482.12889799]

Part 1.B

- **Question**: Present plots of J as a function of the number of iterations for different learning rates. What are good learning rates and number of iterations for this problem?
  Plots: Figure 4.1 shows the plots of J as a function of the number of iterations for different learning rates. Figure 4.2 shows the plots of J when learning rate is 0.3 and the x axis is scaled up.
  Writeup: As we can see in the Figure 4.1, as the learning rate increases, the gradient descent will be faster. However, as it is larger than a threshold, the gradient won't descent any more. Here we find the proper learning rate should be about 0.3. When the learning rate is equal to 0.3, let's scale up the X axis to find the proper number of iterations. As we can see in the Figure 4.2, the proper number of iterations should be around 10, because the gradient remains unchanged after this point.

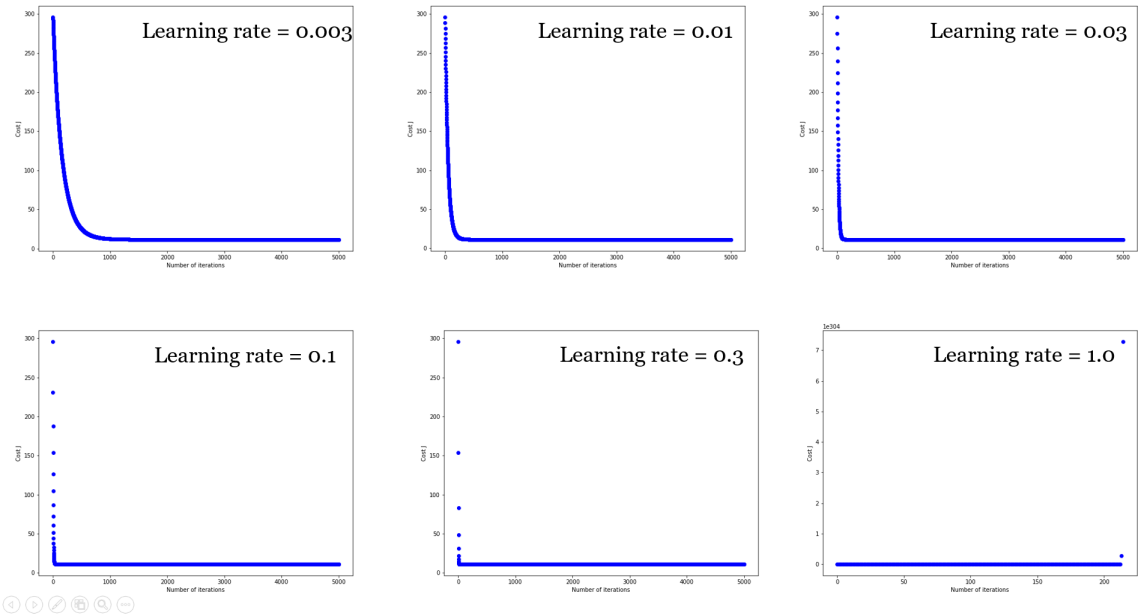


Figure 4.1: learning rates

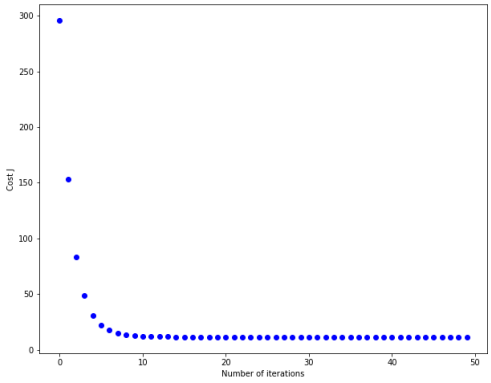
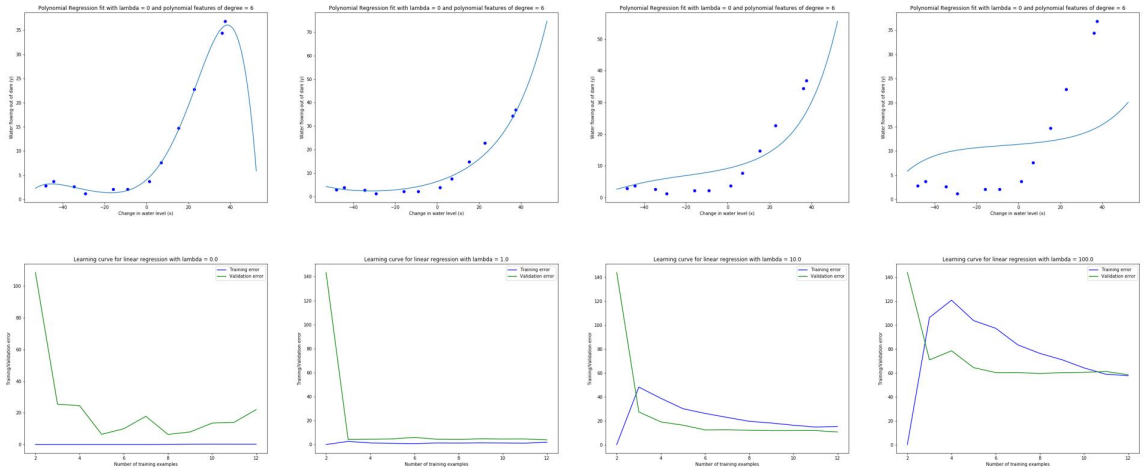


Figure 4.2: learning rates=0.3, scale up x axis



Lambda = 0.0 Lambda = 1.0 Lambda = 10.0 Lambda = 100.0

Figure 4.3: modify lambda

Part 2

- **Question**: Modify the lambda parameter in the cell above and try λ=1,10,100. For each of these values, the script will generate a polynomial fit to the data and also a learning curve. Submit two plots for each value of lambda: the fit as well as the learning curve. Comment on the impact of the choice of lambda on the quality of the learned model.
  Plots: Figure 4.3 shows the plots for each value of lambda: the fit as well as the learning curve. λ=0,1,10,100
  Comment: As λ becomes larger and larger, the model become more and more inaccurate. But the bias becomes lower, and the variance becomes higher. As we can see from the 4.3, the most proper λ here should be around 1.0. At this point, the model fit the data and the gradient will be convergence.

- **Question**: Calculate the error of the best model that you found with the previous analysis and report it.
  Answer:
  when $\lambda = 1.0$, then:
  Optimization terminated successfully.
  Current function value: 6.891076
  Iterations: 21
  Function evaluations: 22
  Gradient evaluations: 22
  3.0987482655574246

- **Question**: You will implement an automated method to select the λ parameter. You should try λ in the following range: 0, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1, 3, 10. Run the cell below to plot a validation curve of λ versus the error. Comment on the best choice

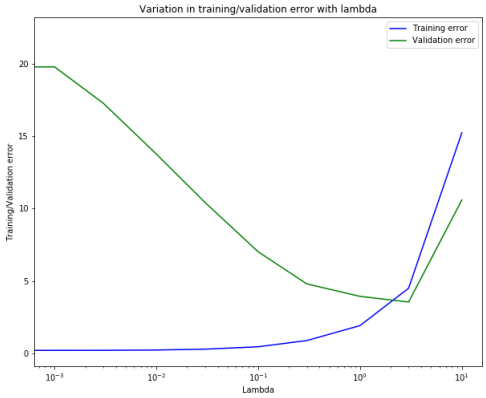


Figure 4.4: validation curve

of λ for this problem.
Plot: Figure 4.4 shows how training error and validation error change as λ increase.
Comment: Taking the error to be minimum when selecting the λ, the best λ is: 3. When the λ is too small, the model tend to be overfit. And when the λ is too big, the model tend to be unfit.

Extra Credit

- **Question**: What is the lowest achievable error on the test set with $\lambda = 0$?
  Answer: Test err of the best linear model with lambda = 0 is: 14.558424000416249.

- **Question**: Select the best value for λ and report the test set error with the best λ?
  Plot: Figure 4.5 shows the training and validation error with different λ.
  Answer: As we can see from the 4.5, the best value for λ is 0. At this time, the lowest achievable error is 14.558424000416249.

- **Question**: Use the technique of adding features to extend each column of the Boston data set with powers of the values in the column. Repeat the bias-variance analysis with quadratic and cubic features. What is the test set error with quadratic features with the best λ chosen with the validation set? What is the test set error with cubic features with the best λ chosen with the validation set?
  Plot: Figure 4.6 shows the training and validation error with different λ when it comes to use the technique of adding features with quadratic, and with cubic features.
  Answer:
  - When it comes to use quadratic features, the best lambda is: 1, and minimum error value is: 5.5471436479140115.
  - When it comes to use cubic features, the the best lambda is: 0.3, and minimum error value is: 7.7848111200607555.

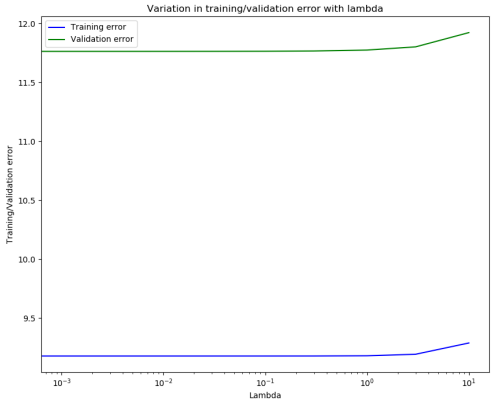


Figure 4.5: modify lambda

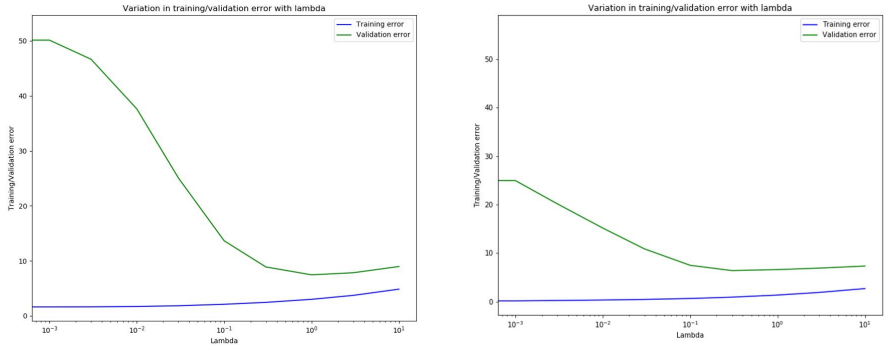


Figure 4.6: modify lambda, with quadratic features (left), and with cubic features (right)

- **Question**: Discuss the impact of regularization for building good models for the Boston housing data set.
  Answer: Regularization allows complex models to be trained on data sets of limited size without severe overfitting, essentially by limiting effective model complexity.