Announcements

Make sure to sign in on the google form (I send a list of which section questions are useful for which pset questions afterwards)

Pset 7 due Friday 11/10

Introductions

- One question or thought related to lecture last week (LASSO, Ridge, cross validation)

Weighted least squares regression

This question is based on a conversation with Skyler Wu.

Consider a least squares model where, rather than weighting all residuals equally, we are going to assign different weights to different residuals. That is, we want to minimize

$$\sum_{i=1}^{n}[w_i(Y_i - \hat{Y}_i)]^2$$

Equivalently, letting $\mathbf{W}$ be a diagonal matrix of weights, letting $\vec{Y} = \mathbf{X}\vec{\beta} + \vec{\epsilon}$ with $\vec{\epsilon} \sim \text{MVN}_n(0, \sigma^2 I_n)$, and using the fact that $\hat{\vec{Y}} = \mathbf{X}\hat{\vec{\beta}}$, we want to minimize

$$||\mathbf{W}(\vec{Y} - \mathbf{X}\hat{\vec{\beta}})||^2$$

Expanding and taking the gradient gives the following:

$$\begin{aligned}
0 &= \frac{\partial}{\partial\hat{\vec{\beta}}}((\vec{Y} - \mathbf{X}\hat{\vec{\beta}})^T\mathbf{W}^T\mathbf{W}(\vec{Y} - \mathbf{X}\hat{\vec{\beta}})) \\
&= -2\mathbf{X}^T\mathbf{W}^T\mathbf{W}(\vec{Y} - \mathbf{X}\hat{\vec{\beta}}) \\
\implies \mathbf{X}^T\mathbf{W}^T\mathbf{W}\mathbf{X}\hat{\vec{\beta}} &= \mathbf{X}^T\mathbf{W}^T\mathbf{W}\vec{Y} \\
\implies \hat{\vec{\beta}} &= (\mathbf{X}^T\mathbf{W}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}^T\mathbf{W}\vec{Y}
\end{aligned}$$

This is our new weighted least-squares regression $\hat{\vec{\beta}}$, which we will be studying in this problem.

Here are a few facts that will be useful here and on the homework:

- For matrices $\mathbf{A}, \mathbf{B}$, and $\mathbf{C}$, of allowable dimensions, $\mathbf{A}(\mathbf{B} + \mathbf{C}) = \mathbf{AB} + \mathbf{AC}$
- If $\mathbf{A}$ is of full column rank, $\mathbf{A}^T\mathbf{A}$ is symmetric and invertible
- If $\mathbf{A}$ is symmetric, and $\mathbf{B}$ is of allowable dimensions, $\mathbf{B}^T\mathbf{AB}$ is symmetric
- For an invertible and symmetric matrix $\mathbf{A}$, $\mathbf{A}^{-1} = (\mathbf{A}^{-1})^T$
- For $\vec{Y} = \vec{c} + \mathbf{B}\vec{X}$ with $\vec{c}$ and $\mathbf{B}$ constant and $\vec{X}$ random, $E(\vec{Y}) = \vec{c} + \mathbf{B}E(\vec{X})$
- For a vector $\vec{X}$ of length n, $\text{Cov}(\vec{X})$ is an $n \times n$ matrix whose i, j entry is $\text{Cov}(X_i, X_j)$
- For $\vec{Y} = \vec{c} + \mathbf{B}\vec{X}$ with $\vec{c}$ and $\mathbf{B}$ constant and $\vec{X}$ random, $\text{Cov}(\vec{Y}) = \mathbf{B}\text{Cov}(\vec{X})\mathbf{B}^T$

1. Verify that using the usual weights for least squares regression, this formula reduces to the usual estimator for $\hat{\vec{\beta}}$.

In our usual least squares regression, $\mathbf{W} = \mathbf{I}$, so

$$\hat{\vec{\beta}} = (\mathbf{X}^T\mathbf{I}^T\mathbf{I}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{I}^T\mathbf{I}\vec{Y} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\vec{Y}$$

as expected.

2. If $\mathbf{W} = c\mathbf{I}$ for some non-zero constant c, what is $\hat{\vec{\beta}}$? What does this say about when $\mathbf{W}$ is useful?

$$\hat{\vec{\beta}} = (c^2\mathbf{X}^T\mathbf{I}^T\mathbf{I}\mathbf{X})^{-1}c^2\mathbf{X}^T\mathbf{I}^T\mathbf{I}\vec{Y} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\vec{Y}$$

which means the matrix of weights only changes the fit coefficients if the weights are different.

3. Find the bias of $\hat{\vec{\beta}}$ for $\vec{\beta}$.

$$\begin{aligned}
E(\hat{\vec{\beta}}) - \vec{\beta} &= E\left[(\mathbf{X}^T\mathbf{W}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}^T\mathbf{W}\vec{Y}\right] - \vec{\beta} \\
&= (\mathbf{X}^T\mathbf{W}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}^T\mathbf{W}E\left[\vec{Y}\right] - \vec{\beta} \\
&= (\mathbf{X}^T\mathbf{W}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}^T\mathbf{W}\mathbf{X}\vec{\beta} - \vec{\beta} \\
&= \vec{0}
\end{aligned}$$

4. Find the variance-covariance matrix of $\hat{\vec{\beta}}$ in matrix form. What does this reduce to when $\mathbf{W} = \mathbf{I}$? You may find it useful to let $\mathbf{A} = (\mathbf{X}^T\mathbf{W}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}^T\mathbf{W}$ throughout.

$$\begin{aligned}
\text{Cov}(\hat{\vec{\beta}}) &= \text{Cov}\left[\mathbf{A}\vec{Y}\right] \\
&= \mathbf{A}\text{Cov}\left(\vec{Y}\right)\mathbf{A}^T \\
&= \mathbf{A}\text{Cov}\left(\vec{\epsilon}\right)\mathbf{A}^T \\
&= \sigma^2\mathbf{A}\mathbf{I}\mathbf{A}^T \\
&= \sigma^2\mathbf{A}\mathbf{A}^T
\end{aligned}$$

When $\mathbf{W}^T\mathbf{W} = I$, $\mathbf{A} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$, so

$$\text{Cov}(\hat{\vec{\beta}}) = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1} = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}$$

which is the usual variance-covariance matrix for linear regression.

Ridge, LASSO, optimizing λ, and β trajectories

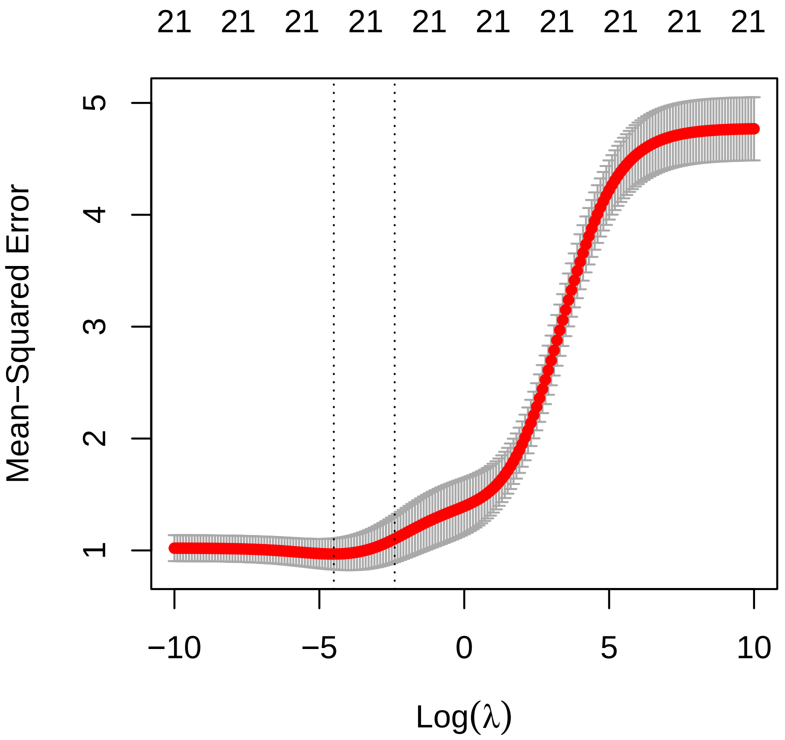
These problems will deal with a dataset of country-level statistics from UNdata, Varieties of Democracy, and the World Bank.

1. First, we'll fit a well-tuned Ridge regression model via `cv.glmnet` for predicting log2 GDP per capita from a country's urban population, its proportion of people 60+, its arable land, its energy supply, its unemployment rate, and its number of tourists and visitors. We'll perform 10-fold cross validation to choose the optimal λ. What is the optimal model?

```
##
## Call:  cv.glmnet(x = X, y = y, lambda = exp(seq(-10, 10, 0.1)), nfolds = 10,      alpha = 0)
##
## Measure: Mean-Squared Error
##
##      Lambda Index Measure     SE Nonzero
## min 0.01111   146   0.968 0.1390      21
## 1se 0.09072   125   1.103 0.2071      21
```

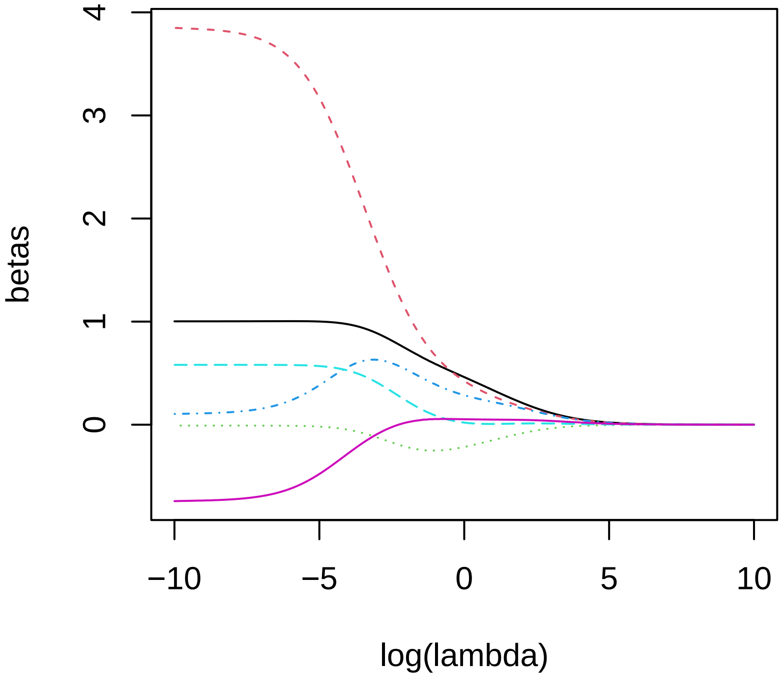
The best ridge regression model is one with a λ of 0.011. A λ up to 0.091 might also be reasonable.

2. The following is a plot of MSE on the validation sets against the λ's from the previous part. Justify the previous λ with this plot.



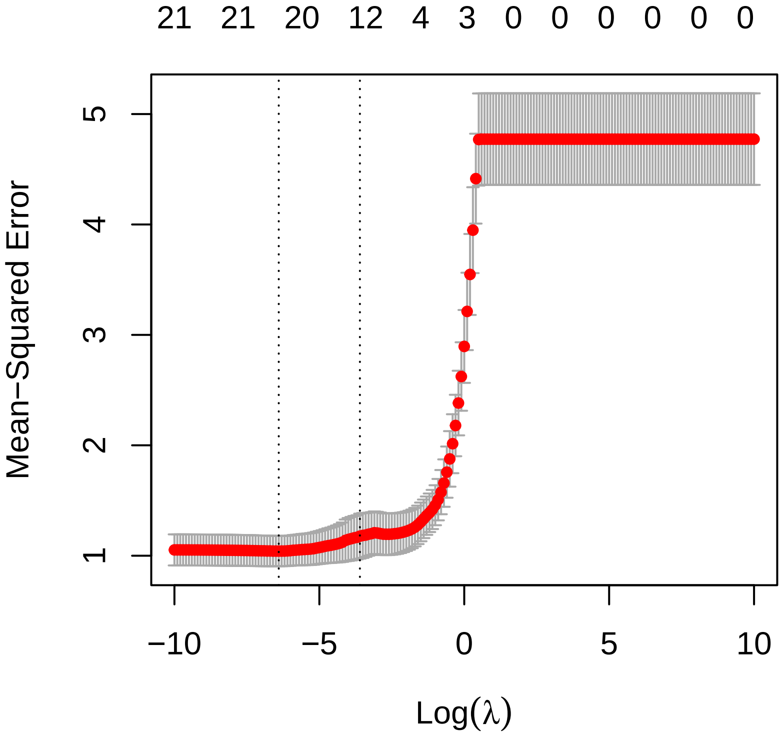
The best λ is 0.011, which corresponds to $\log(0.011) = -4.51$ in the plot. Values of λ slightly above this might also help in preventing overfitting.

3. The following are the $\hat{\beta}$ trajectories of the main (non-interaction) effects from this model. Interpret what you see in 2-3 sentences.



We can see that the estimates vary a lot for low values of λ, but coefficients shrink to zero asymptotically. However, as λ grows, the $\hat{\beta}$s never reach zero exactly as expected in ridge regression.

4. Next, we'll fit a well-tuned LASSO regression model for the same question. We'll perform 10-fold cross validation to choose the optimal λ. What is the optimal model? Is this consistent with the plot?

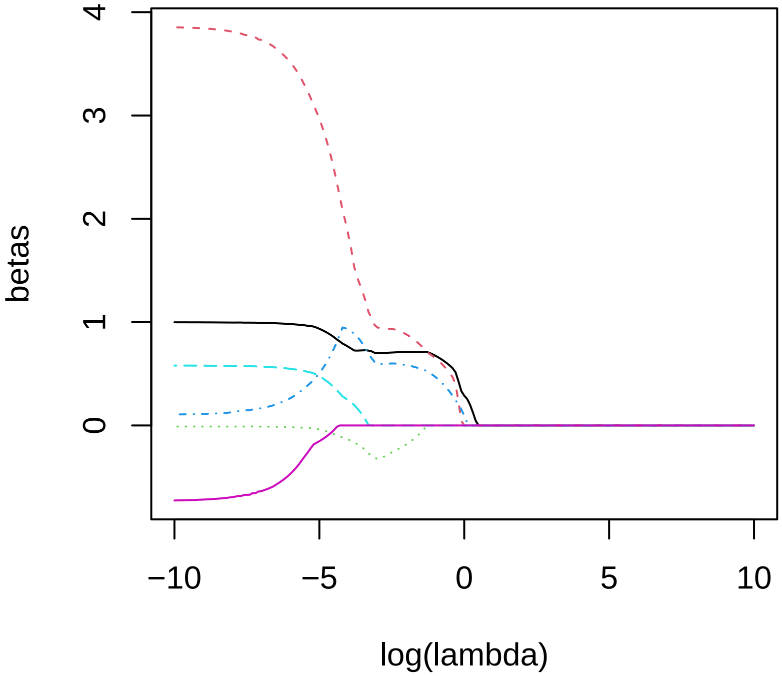


```
##
## Call:  cv.glmnet(x = X, y = y, lambda = exp(seq(-10, 10, 0.1)), nfolds = 10,      alpha = 1)
##
## Measure: Mean-Squared Error
```

```
##
##       Lambda Index Measure     SE Nonzero
## min 0.001662   165   1.042 0.1374      21
## 1se 0.027324   137   1.177 0.2037      10
```

The best λ is 0.00166, which corresponds to $\log(0.00166) = -6.4$ in the plot. Values of λ slightly above this might also help in preventing overfitting.

5. The following are the $\hat{\beta}$ trajectories of the main (non-interaction) effects from this model. Compare these to the ridge trajectories.



The estimates vary a lot for low values of λ, and they even increase in magnitude occasionally, indicating collinearity. However, as λ grows, more and more snap to 0. Compared to ridge regression, these coefficients decrease more sporadically but actually become 0 rather than just approaching 0.

6. What is the best regularized/penalized regression model?

```
##     Ridge, lambda=0.011 LASSO, lambda=0.0017
## MSE           0.9679976              1.04198
```

These are the minimum means of cross validated error (the cross validated MSE using the optimal λ). Ridge has a slightly lower MSE on cross-validation, so ridge with $\lambda = 0.011$ is the best regularized model.

Penalization functions

Recall that for both Ridge and LASSO, we are trying to minimize something of the form:

$$\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 + p(\vec{\hat{\beta}})$$

State whether the following functions could or couldn't be used as penalization functions for $\vec{\hat{\beta}}$. If so, provide a context in which this might be a useful penalization function; if not, explain why it would give undesired behavior.

1. $p(\vec{\hat{\beta}}) = \sum_{i=1}^{k} \hat{\beta}_i$

No: the $\hat{\beta}_i$ would just get more and more negative.

2. $p(\vec{\hat{\beta}}) = \sum_{i=1}^{k} \hat{\beta}_i{}^4$

Yes: you could use this if you wanted to very strongly penalize $\hat{\beta}$s with large magnitudes.

3. $p(\vec{\hat{\beta}}) = \sum_{i=1}^{k} \log(\hat{\beta}_i)$

No: negative $\hat{\beta}_i$ would break this.

4. $p(\vec{\hat{\beta}}) = \sum_{i=1}^{k} \log(|\hat{\beta}_i|)$

No: the $\hat{\beta}_i$ would always go to 0 since that would give $-\infty$ loss.

5. $p(\vec{\hat{\beta}}) = \sum_{i=1}^{k} 1/|\hat{\beta}_i|$

No: larger $\hat{\beta}$s give a smaller loss.

6. $p(\vec{\hat{\beta}}) = -\sum_{i=1}^{k} 1/|\hat{\beta}_i|$

Still no: the $\hat{\beta}_i$ would always go to 0 since that would give $-\infty$ loss.

7. What general requirements do we need for a penalization function?

The loss should increase (or remain 0) with coefficients of increasing magnitude, and it should have a (preferably global) minimum.

8. Write a valid penalization function that we haven't studied before.

One example would be

$$p(\vec{\hat{\beta}}) = \sum_{i=1}^{k} f(\hat{\beta}_i) \text{ with } f(\hat{\beta}_i) = \begin{cases} 0 & |\hat{\beta}_i| \leq \delta \\ |\hat{\beta}_i| & \text{Otherwise} \end{cases}$$

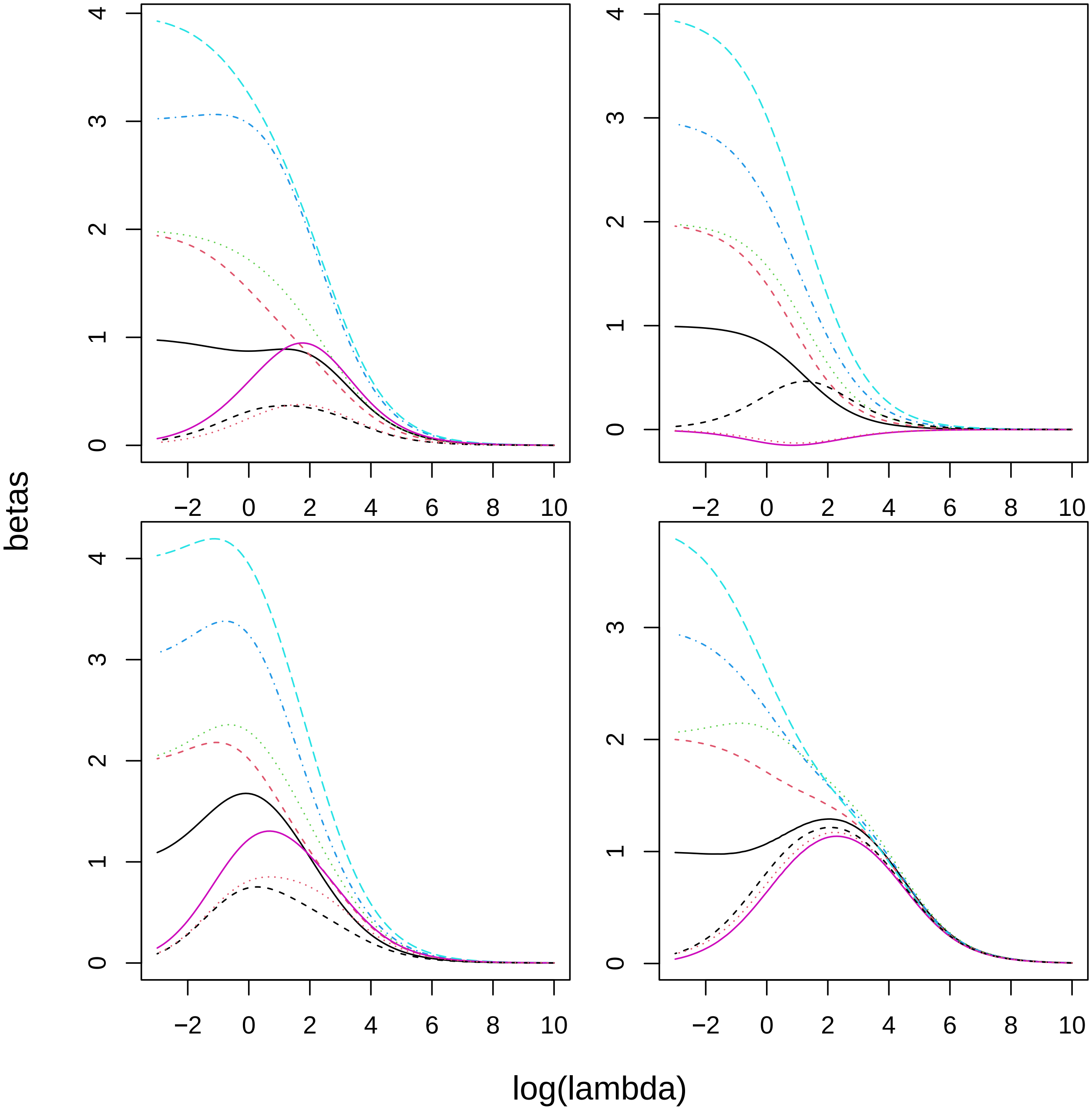
for some δ, which applies no penalty once a $\hat{\beta}_i$ is within δ of 0.

Miscellaneous

1. For what λs would LASSO and Ridge give the same model?

$\lambda = 0, \infty$ since there would be either no penalization or the model would just be $\bar{Y}$.

2. Below are four $\hat{\beta}$ trajectory plots for ridge regressions. Each comes from a data set with 50 data points. One trajectory comes from data with no built-in correlation between the predictors; one comes from data with moderate and equal correlation among all the predictors; one comes from data with moderate random (but fixed) correlation among the predictors; and one is fake (and impossible). Determine which is which.



The first plot is the random correlation as seen by its peaks rising and falling more sporadically. The second is the no correlation plot because the slopes mostly fall independently of each other. The third is impossible because all the slopes rise at the beginning (but at least one must fall for the other to rise). The fourth is the strong correlation as seen by the fact that all the slopes are about the same after $\lambda = e^2$.