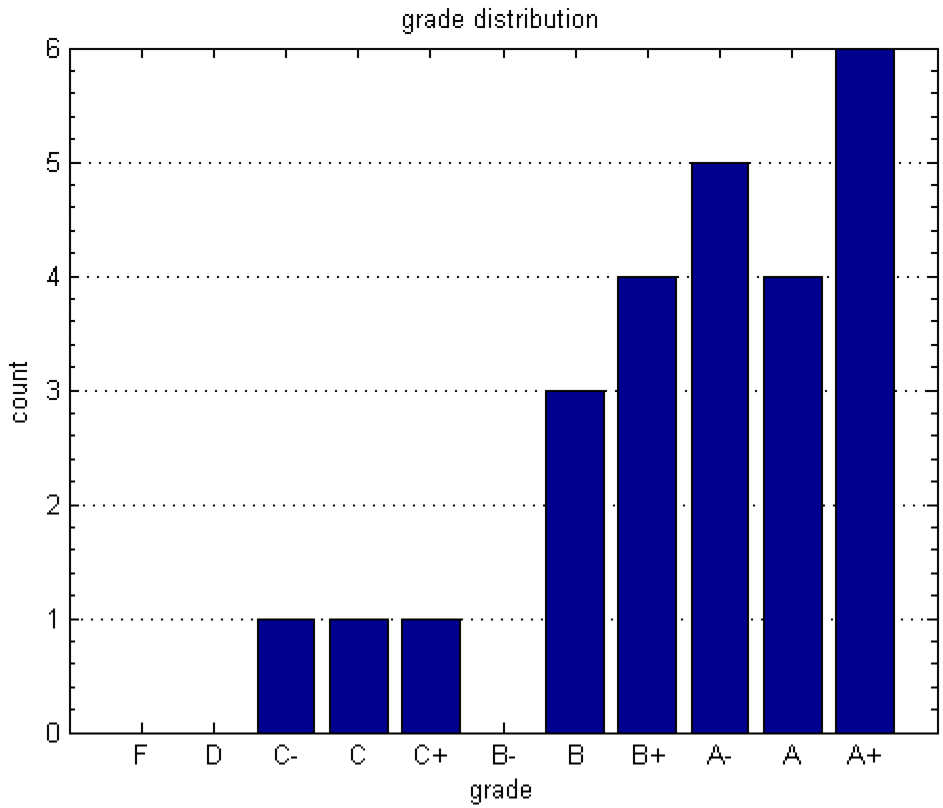
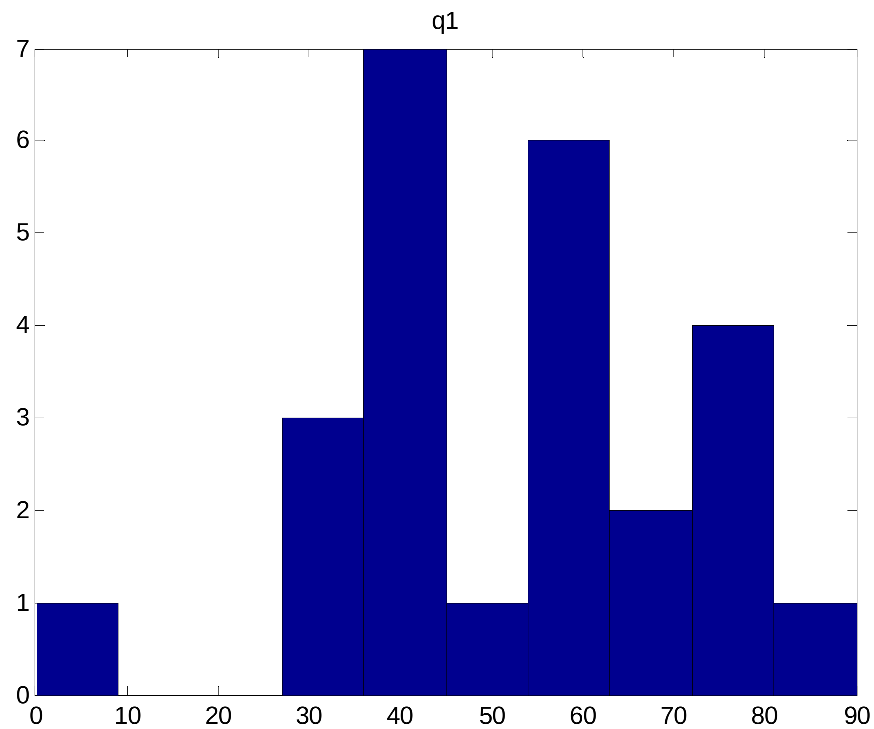
CS540 Machine learning
Lecture 12
Feature selection

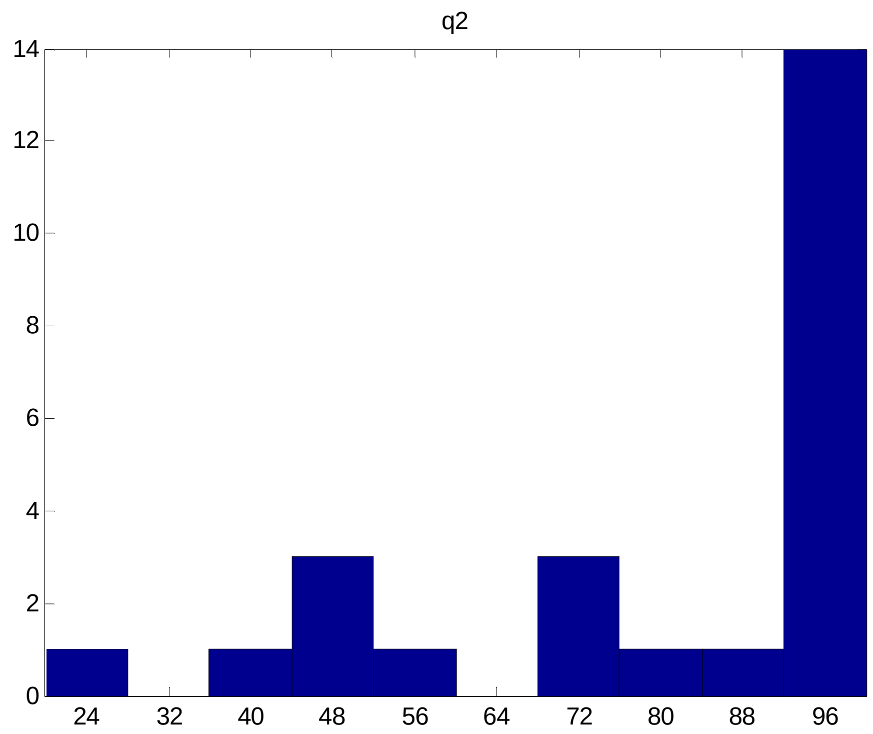
Midterm



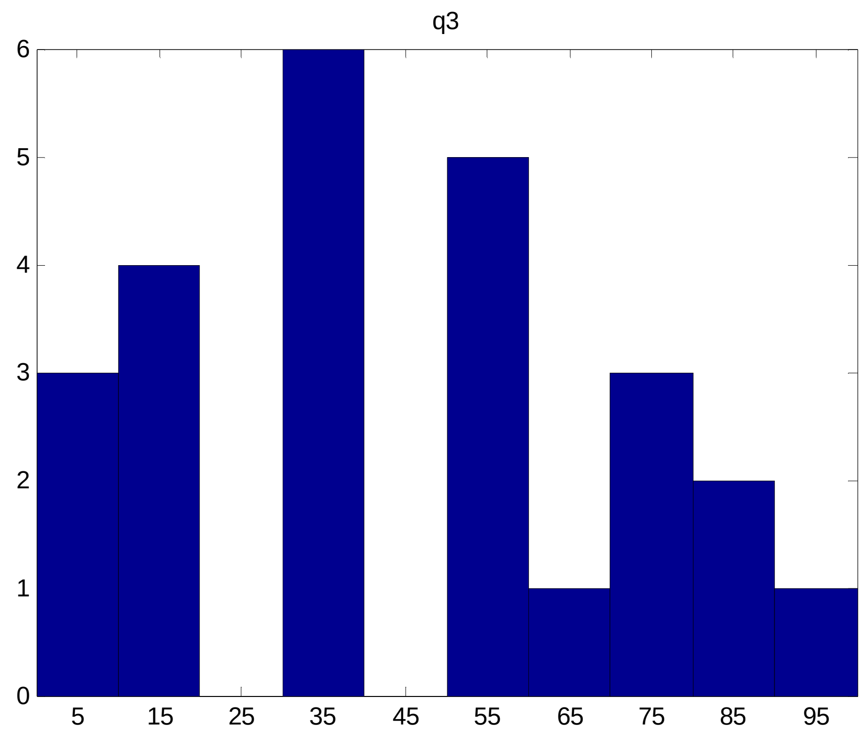
Q1



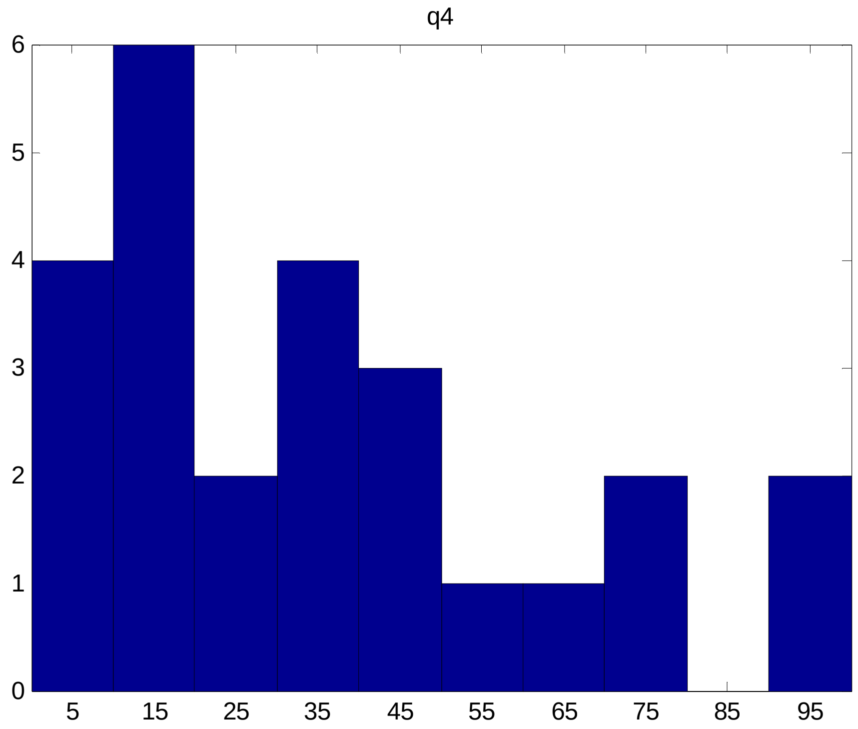
Q2



Q3



Q4



Outline

- Problem formulation
- Filter methods
- Wrapper methods
- L1 methods

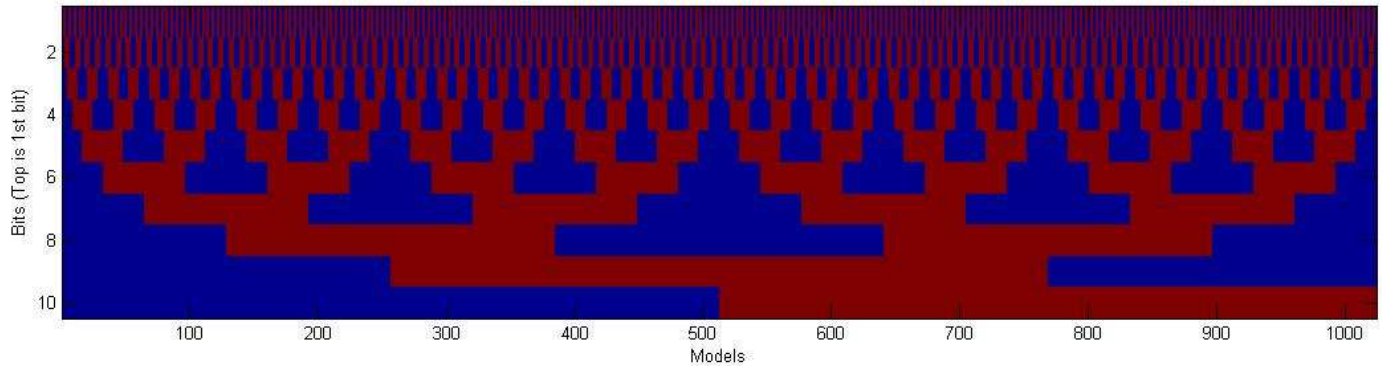
Feature selection

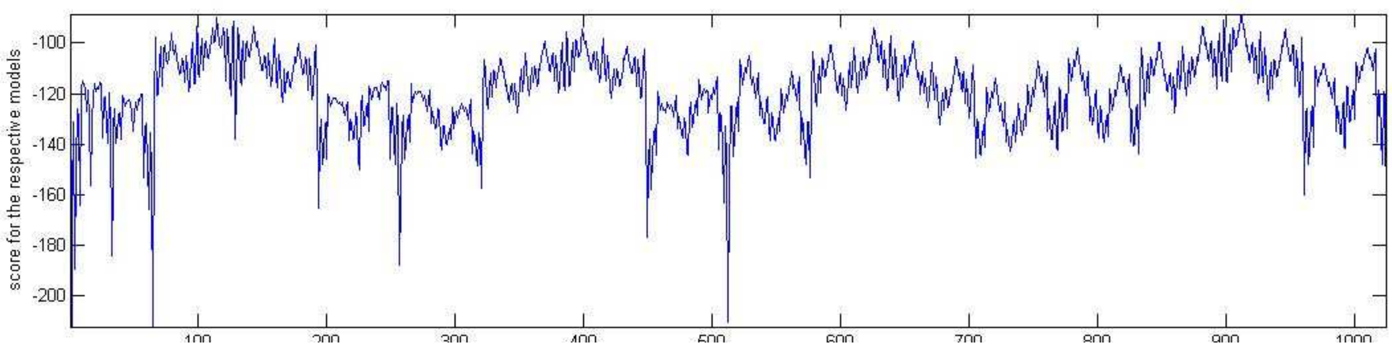
- If predictive accuracy is the goal, often best to keep all predictors and use L2 regularization
- We often want to select a subset of the inputs that are “most relevant” for predicting the output, to get sparse models – interpretability, speed, possibly better predictive accuracy

Bayesian formulation

- Let m specify which of the 2^d subsets of variables to use (bit vector)

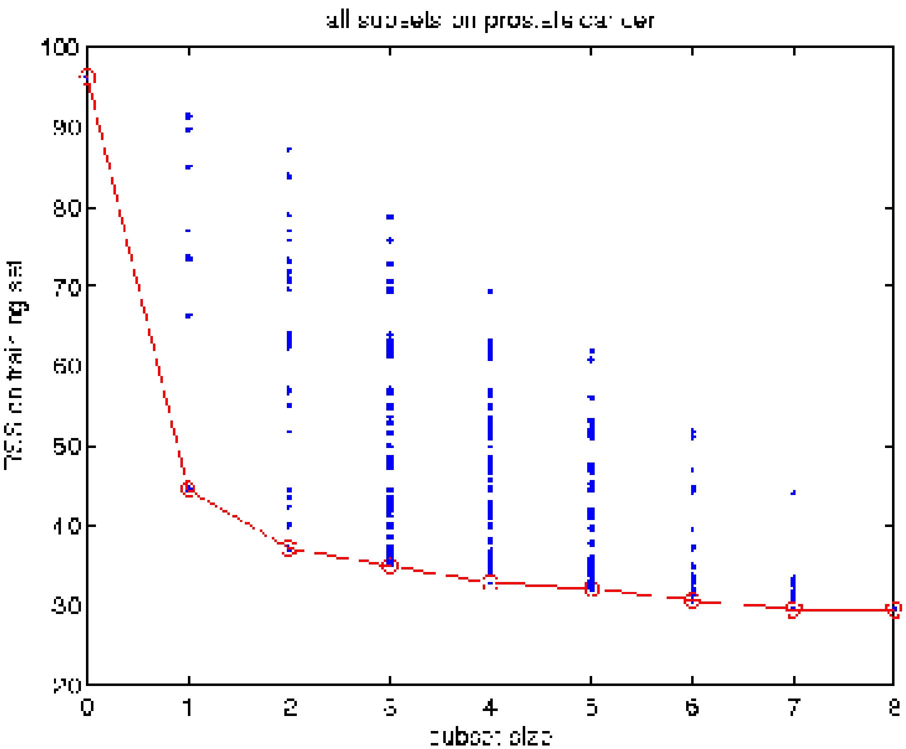
$$
\begin{aligned}
p(m|\mathcal{D}) &\propto p(\mathcal{D}|m)p(m) \\
p(\mathcal{D}|m) &= \int \prod_i p(y_i|\mathbf{x}_i, \mathbf{w}, m)p(\mathbf{w}|m)d\mathbf{w}
\end{aligned}
$$





Statistical problem

- What if we cannot evaluate marginal likelihood p(D|m)?
- Cannot use MLE since will always pick largest subset



Penalized likelihood

- Common to pick the model that minimizes

$$J(m) = -\log p(\mathcal{D}|m) + \lambda \text{complexity}(m)$$

- Eg complexity(m) = #chosen variables
- For linear regression

$$J(m) = RSS(\mathbf{w}) + \lambda||\mathbf{w}||_0, \quad \mathbf{w} = (\mathbf{X}(:,m)^T\mathbf{X}(:,m))^{-1}\mathbf{X}(:,m)^T\mathbf{y}$$

Computational problem

- 2^d subsets to evaluate

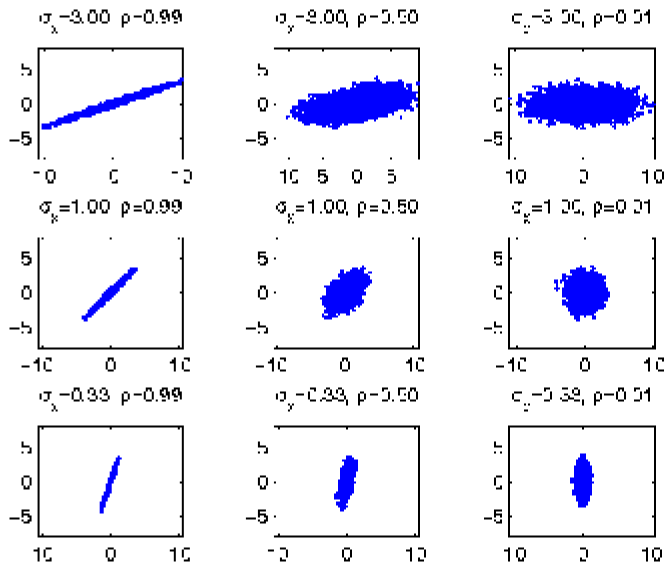
Filter methods

- Compute "relevance" of Xj to Y marginally
- Computationally efficient

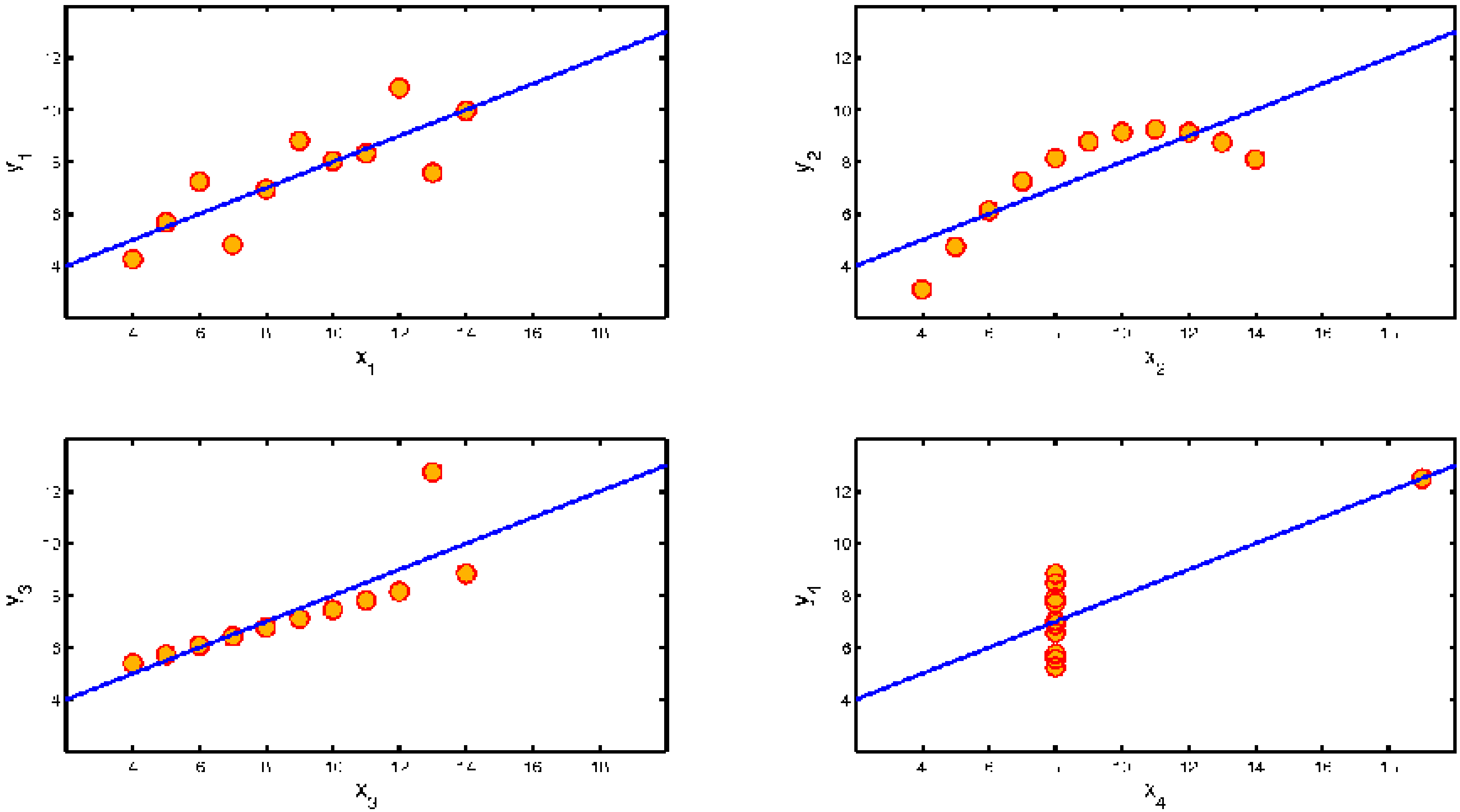
Correlation coefficient

- Measures extent to which X_j and Y are linearly related

$$\rho_{X_j,Y} = \frac{\text{Cov}(X_j, Y)}{\sqrt{\text{Var}\ (X_j)\text{Var}\ (Y)}}$$



Anscombe's quartet



ρ=0.81

Mutual information

- Can model non linear non Gaussian dependencies

$$I(X_j, Y) = \int \int p(x_j, y) \log \frac{p(x_j, y)}{p(x_j)p(y)} dx_j dy$$

- If assume p(X,Y) is Gaussian, recover correlation coef. Can use non-parametric density estimates to get better estimate.
- For discrete data, can estimate p(X,Y) by counting.

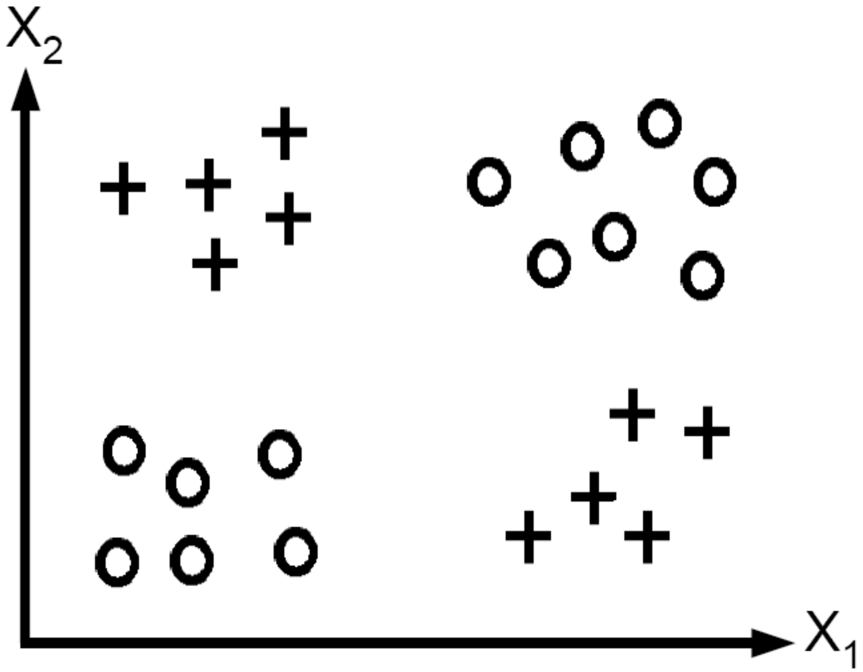
$$\begin{aligned} I(X_j, Y) &= \sum_{x_j} \sum_{y} p(x_j, y) \log \frac{p(x_j, y)}{p(x_j)p(y)} \\ \hat{p}(x_j = a, y = b) &= \frac{\sum_i I(x_{ij} = a, y = b)}{n} \end{aligned}$$

MI for NB with binary features

$$
\begin{aligned}
I(X_j, Y) &= \sum_{x=0}^{1} \sum_{c=1}^{C} p(X_j = x, y = c) \log \frac{p(X_j = x|y = c)p(y = c)}{p(X_j = x)p(y = c)} \\
&= \sum_{x=0}^{1} \sum_{c} p(X_j = x|y = c)p(y = c) \log \frac{p(X_j = x|y = c)}{p(X_j = x)} \\
&= \sum_{c} p(X_j = 1|y = c)p(y = c) \log \frac{p(X_j = 1|y = c)}{p(X_j = 1)} \\
&\qquad + \sum_{c} p(X_j = 0|y = c)p(y = c) \log \frac{p(X_j = 0|y = c)}{p(X_j = 0)} \\
&= \sum_{c} \left[\theta_{jc}\pi_c \log \frac{\theta_{jc}}{\theta_j} + (1 - \theta_{jc})\pi_c \log \frac{1 - \theta_{jc}}{1 - \theta_j} \right]
\end{aligned}
$$

What's wrong with filter methods

- Interaction effects (eg SNPs)



Wrapper methods

- Perform discrete search in model space
- “Wrap” search around standard model fitting
- Forwards selection, backwards selection, heuristic algorithms (GAs, SLS, SA, etc)
- Need efficient way to evaluate score of models m’ in neighborhood of m

{1,2,3,4}

{1,2,3} {2,3,4} {1,3,4} {1,2,4}

{1,2} {1,3} {1,4} {2,3} {2,4} {3,4}

{1} {2} {3} {4}

{}

Forward selection for linear regression

- At each step, add feature that maximally reduces residual error.
- If choose j, should set its weight to be the orthogonal projection of r onto column j

$$
\begin{aligned}
J(w_j) &= ||\mathbf{r} - \mathbf{x}_j w_j||_2^2 = \mathbf{r}^T\mathbf{r} + w_j^2 \mathbf{x}_j^T \mathbf{x}_j - 2 w_j \mathbf{x}_j^T \mathbf{r} \\
\frac{dJ}{dw_j} &= 0 \qquad \text{homework} \\
\hat{w}_j &= \frac{\mathbf{x}_j^T \mathbf{r}}{\mathbf{x}_j^T \mathbf{x}_j}
\end{aligned}
$$

Choosing the best feature

- Inserting formula for optimal w_j

$$
\begin{aligned}
J(\hat{w}_j) &= \mathbf{r}^T\mathbf{r} + \frac{(\mathbf{x}_j^T\mathbf{r})^2}{\mathbf{x}_j^T\mathbf{x}_j} - 2\frac{(\mathbf{x}_j^T\mathbf{r})^2}{\mathbf{x}_j^T\mathbf{x}_j} = \mathbf{r}^T\mathbf{r} - \frac{(\mathbf{x}_j^T\mathbf{r})^2}{\mathbf{x}_j^T\mathbf{x}_j} \\
k &= \arg\min_j J(\hat{w}_j) = \arg\max_j \frac{(\mathbf{x}_j^T\mathbf{r})^2}{\mathbf{x}_j^T\mathbf{x}_j}
\end{aligned}
$$

- If features are unit norm, we pick j with largest inner product (smallest angle) to r

$$
k = \arg\min_j J(\hat{w}_j) = \arg\max_j (\mathbf{x}_j^T\mathbf{r})^2
$$

Orthogonal least squares

- Once chosen k, project onto subspace orthogonal to 1:k

Algorithm 1: Forward stepwise selection (Orthogonal least squares)

$\mathbf{r} \leftarrow \mathbf{y}$, used $\leftarrow \emptyset$, unused $\leftarrow 1$ **to** n
repeat
    $k \leftarrow \arg\max_{j \in \text{unused}} \mathbf{x}_j^T \mathbf{r}$
    $\mathbf{r} \leftarrow \mathbf{r} - (\mathbf{x}_k^T \mathbf{r})\mathbf{x}_k$
    move k from unused to used
    foreach $j \in$ unused **do**
        $\mathbf{x}_j \leftarrow \mathbf{x}_j - (\mathbf{x}_j^T \mathbf{x}_k)\mathbf{x}_k$
        $\mathbf{x}_j \leftarrow \mathbf{x}_j / ||\mathbf{x}_j||$
until *stopping criterion is met*

L1 is convex relaxation of L0

- For linear regression

$$J_0(m) = RSS(\mathbf{w}) + \lambda ||\mathbf{w}||_0$$

$$||\mathbf{w}||_0 \quad = \quad \sum_{j=1}^{d} I(|w_j| > 0)$$

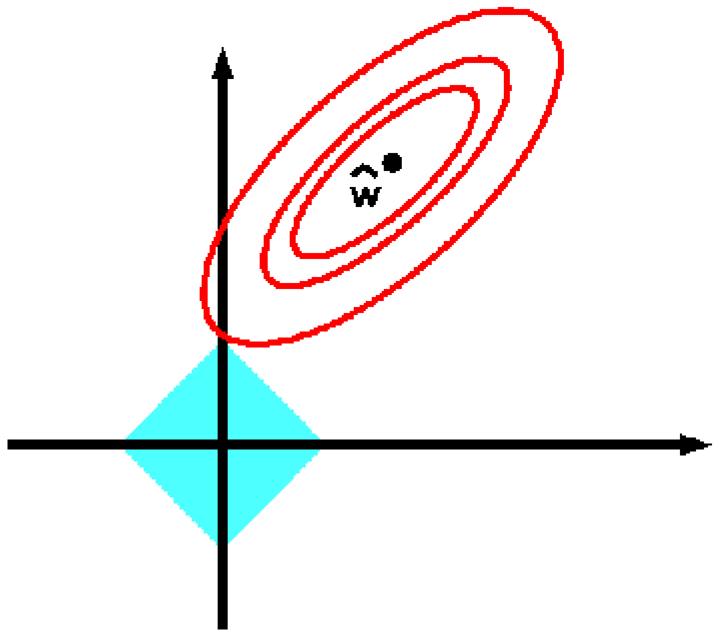
$$J_1(m) = RSS(\mathbf{w}) + \lambda ||\mathbf{w}||_1$$

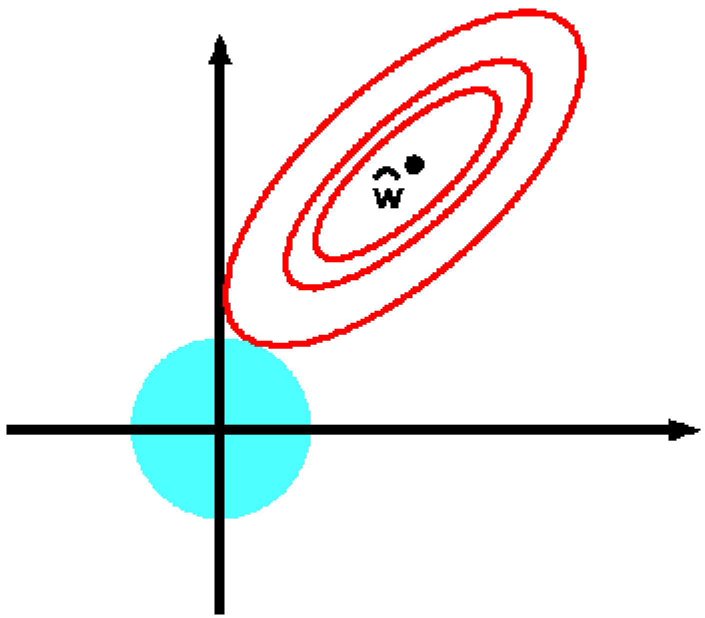
$$||\mathbf{w}||_1 \quad = \quad \sum_{j=1}^{d} |w_j|$$

Lasso

$$J(\mathbf{w}) = RSS(\mathbf{w}) + \lambda||\mathbf{w}||_1$$

$$J(\mathbf{w}) = RSS(\mathbf{w}) + \lambda||\mathbf{w}||_2^2$$





Whence sparsity?

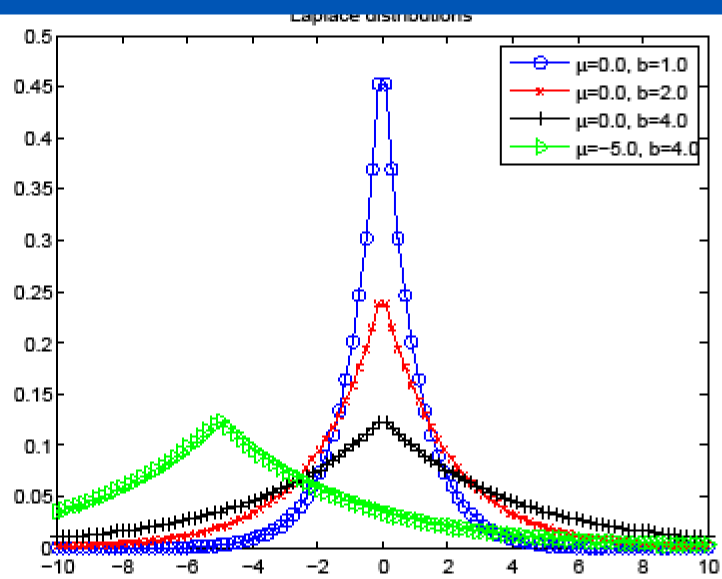
- Ridge prior: all points on unit circle equal under the prior

$$||(1,0)||_2 = ||(1/\sqrt{2}, 1/\sqrt{2}||_2 = 1$$

- Lasso prior: points on corner of simples more probable a priori

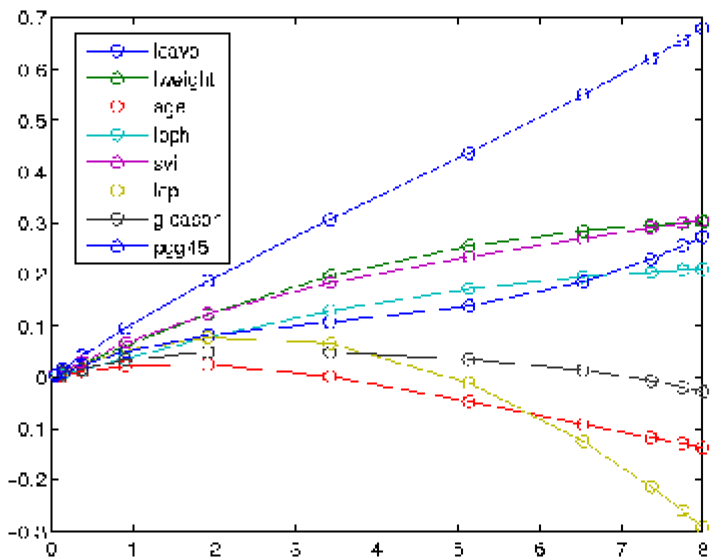
$$||(1,0)||_1 = 1 < ||(1/\sqrt{2}, 1/\sqrt{2}||_1 = \sqrt{2}$$

Lasso as MAP estimation

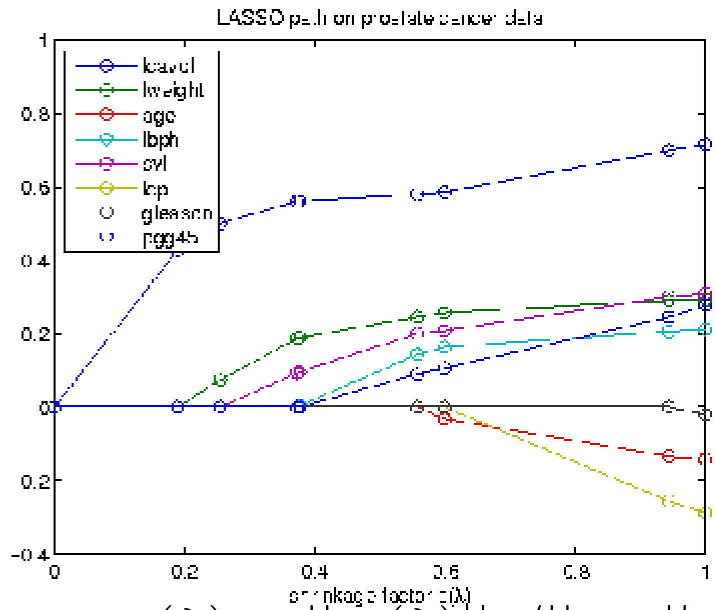


$$
\begin{aligned}
p(\mathbf{w}) &= \prod_{j=1}^{d} DE(w_j|0,\tau) \\
DE(w_j|\mu,\tau) &= \frac{1}{2\tau}\exp\left(-\frac{|w_j-\mu|}{\tau}\right) \\
\hat{\mathbf{w}} &= \arg\max_{\mathbf{w}} \log p(\mathbf{w}|\mathcal{D}) = \arg\max_{\mathbf{w}} \log p(\mathbf{w}) + \log p(\mathcal{D}|\mathbf{w}) \\
&= \arg\max_{\mathbf{w}} -\frac{1}{\tau}\sum_{j=1}^{d}|w_j| - \frac{1}{2\sigma^2}||\mathbf{y}-\mathbf{X}\mathbf{w}||_2^2 \\
\hat{\mathbf{w}} &= \arg\min_{\mathbf{w}} RSS(\mathbf{w}) + \lambda||\mathbf{w}||_1 \\
\lambda &\overset{\text{def}}{=} \frac{2\sigma^2}{\tau}
\end{aligned}
$$

Regularization path



dof(λ)



$$s(\lambda) = ||\mathbf{w}(\lambda)||_1 / ||\mathbf{w}_{ls}||_1$$

Listing 1: :

0	0	0	0	0	0	0	0
0.4279	0	0	0	0	0	0	0
0.5015	0.0735	0	0	0	0	0	0
0.5610	0.1878	0	0	0.0930	0	0	0
0.5622	0.1890	0	0.0036	0.0963	0	0	0
0.5797	0.2456	0	0.1435	0.2003	0	0	0.0901
0.5864	0.2572	-0.0321	0.1639	0.2082	0	0	0.1066
0.6994	0.2910	-0.1337	0.2062	0.3003	-0.2565	0	0.2452
0.7164	0.2926	-0.1425	0.2120	0.3096	-0.2890	-0.0209	0.2773

Lambda max

- Lambda=0 is OLS/MLE
- Max value sets all weights to 0

$$J(\mathbf{w}) = RSS(\mathbf{w}) + \lambda||\mathbf{w}||_1$$

$$\lambda_{max} = ||2\mathbf{X}^T\mathbf{y}||_\infty = 2\max_j |\mathbf{y}^T\mathbf{x}_{:,j}|$$

Homework