CPSC 340: Machine Learning and Data Mining

Regularization

Fall 2017

Admin

- Assignment 2
  - 2 late days to hand in tonight, answers posted tomorrow morning.
- Extra office hours
  - Thursday at 4pm (ICICS 246).
- Midterm details:
  - Friday in class, details on Piazza.

Last Time: Feature Selection

- Last time we discussed feature selection:
  - Choosing set of “relevant” features.

$$X = \begin{bmatrix} \quad & \quad & \quad \end{bmatrix} \qquad y = \begin{bmatrix} \quad \end{bmatrix}$$

“relevant”

- Most common approach is search and score:
  - Define “score” and “search” for features with best score.
- But it’s hard to define the “score” and it’s hard to “search”.
  - So we often use greedy methods like forward selection.
- Methods work ok on “toy” data, but are frustrating on real data…

Is “Relevance” Clearly Defined?

- Consider a supervised classification task:

gender	mom	dad
F	1	0
M	0	1
F	0	0
F	1	1

SNP
1
0
0
1

- Predict whether someone has particular genetic variation (SNP).
  - Location of mutation is in “mitochondrial” DNA.
    - “You almost always have the same value as your mom”.

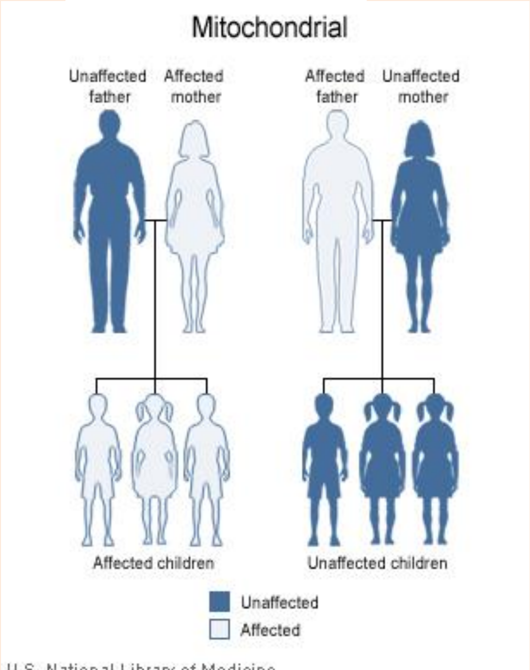
Is “Relevance” Clearly Defined?

- Consider a supervised classification task:

gender	mom	dad
F	1	0
M	0	1
F	0	0
F	1	1

SNP
1
0
0
1

- True model:
  - (SNP = mom) with very high probability.
  - (SNP != mom) with some very low probability.
- What are the “relevant” features for this problem?
  - Mom is relevant and {gender, dad} are not relevant.



https://en.wikipedia.org/wiki/Human_mitochondrial_genetics

Is “Relevance” Clearly Defined?

- What if “mom” feature is repeated?

gender	mom	dad	mom2
F	1	0	1
M	0	1	0
F	0	0	0
F	1	1	1

SNP
1
0
0
1

- Are “mom” and “mom2” relevant?
  - Should we pick them both?
  - Should we pick one because it predicts the other?

Neither of these is “correct”, but not picking either is incorrect.

- General problem (“dependence”, “collinearity” for linear models):
  - If features can be predicted from features, don’t know one(s) to pick.

Is "Relevance" Clearly Defined?

- What if we add (maternal) "grandma"?

gender	mom	dad	grandma
F	1	0	1
M	0	1	0
F	0	0	0
F	1	1	1

SNP
1
0
0
1

- Is "grandma" relevant?
  - You can predict SNP very accurately from "grandma" alone.
  - But "grandma" is irrelevant if I know "mom".
- General problem (conditional independence):
  - "Relevant" features may be irrelevant given other features.

Is “Relevance” Clearly Defined?

- What if we don’t know “mom”?

gender	grandma	dad
F	1	0
M	0	1
F	0	0
F	1	1

SNP
1
0
0
1

- Now is “grandma” is relevant?
  - Without “mom” variable, using “grandma” is the best you can do.
- General problem (“taco Tuesday”):
  - Features can be relevant due to missing information.

Is “Relevance” Clearly Defined?

- What if we don’t know “mom” or “grandma”?

gender	dad
F	0
M	1
F	0
F	1

SNP
1
0
0
1

- Now there are no relevant variables, right?
  – But “dad” and “mom” must have some common maternal ancestor.
  – “Mitochondrial Eve” estimated to be ~200,000 years ago.
- General problem (effect size):
  – “Relevant” features may have small effects.

Is “Relevance” Clearly Defined?

- What if we don’t know “mom” or “grandma”?

gender	dad
F	0
M	1
F	0
F	1

SNP
1
0
0
1

- Now there are no relevant variables, right?
  – What if “mom” likes “dad” because he has the same SNP as her?
- General problem (confounding):
  – Hidden effects can make “irrelevant” variables “relevant”.

Is "Relevance" Clearly Defined?

- What if we add "sibling"?

gender	dad	sibling
F	0	1
M	1	0
F	0	0
F	1	1

SNP
1
0
0
1

- Sibling is "relevant" for predicting SNP, but it's not the cause.
- General problem (non-causality or reverse causality):
  - A "relevant" feature may not be causal, or may be an effect of label.

Is “Relevance” Clearly Defined?

- What if don’t have “mom” but we have “baby”?

gender	dad	baby
F	0	1
M	1	1
F	0	0
F	1	1

SNP
1
0
0
1

- “Baby” is relevant when (gender == F).
  - “Baby” is relevant (though causality is reversed).
  - Is “gender” relevant?
    - If we want to find relevant causal factors, “gender” is not relevant.
    - If we want to predict SNP, “gender” is relevant.
- General problem (context-specific relevance):
  - Adding a feature can make an “irrelevant” feature “relevant”.

Is “Relevance” Clearly Defined?

- Warnings about feature selection:
  - A feature is only “relevant” in the context of available features.
    - Adding/removing features can make features relevant/irrelevant.
  - Confounding factors can make “irrelevant” variables the most “relevant”.
  - If features can be predicted from features, you can’t know which to pick.
    - Collinearity is a special case of “dependence” (which may be non-linear).
  - A “relevant” feature may have a tiny effect.
  - “Relevance” for prediction does not imply a causal relationship.

Is this hopeless?

- We often want to do feature selection we so have to try!

- Different methods are affected by problems in different ways.
  - We'll ignore causality and confounding issues (bonus slides).

- These "problems" don't have right answers but have wrong answers:
  - Variable dependence ("mom" and "mom2" have same information).
  - Conditional independence ("grandma" is irrelevant given "mom").

- These "problems" have application-specific answers:
  - Tiny effects.
  - Context-specific relevance (is "gender" relevant if given "baby"?).

Rough Guide to Feature Selection

Method\Issue	Dependence	Conditional Independence	Tiny effects	Context-Specific Relevance
Association (e.g., measure correlation between features 'j' and 'y')	Ok (takes "mom" and "mom2")	Bad (takes "grandma", "great-grandma", etc.)	Ignores	Bad (misses features that must interact, "gender" irrelevant given "baby")

Rough Guide to Feature Selection

Method\Issue	Dependence	Conditional Independence	Tiny effects	Context-Specific Relevance
Association (e.g., measure correlation between features 'j' and 'y')	Ok (takes "mom" and "mom2")	Bad (takes "grandma", "great-grandma", etc.)	Ignores	Bad (misses features that must interact, "gender" irrelevant given "baby")
Regression Weight (fit least squares, take biggest $\|w_j\|$)	Bad (can take irrelevant but collinear, can take none of "mom1-3")	Ok (takes "mom" not "grandma", if linear and 'n' large.	Ignores (unless collinear)	Ok (if linear, "gender" relevant give "baby")

Rough Guide to Feature Selection

Method\Issue	Dependence	Conditional Independence	Tiny effects	Context-Specific Relevance
Association (e.g., measure correlation between features 'j' and 'y')	Ok (takes "mom" and "mom2")	Bad (takes "grandma", "great-grandma", etc.)	Ignores	Bad (misses features that must interact, "gender" irrelevant given "baby")
Regression Weight (fit least squares, take biggest $\|w_j\|$)	Bad (can take irrelevant but collinear, can take none of "mom1-3")	Ok (takes "mom" not "grandma", if linear and 'n' large.	Ignores (unless collinear)	Ok (if linear, "gender" relevant give "baby")
Search and Score w/ Validation Error	Ok (takes at least one of "mom" and "mom2")	Bad (takes "grandma", "great-grandma", etc.)	Allows	Ok ("gender" relevant given "baby")

Rough Guide to Feature Selection

Method\Issue	Dependence	Conditional Independence	Tiny effects	Context-Specific Relevance
Association (e.g., measure correlation between features 'j' and 'y')	Ok (takes "mom" and "mom2")	Bad (takes "grandma", "great-grandma", etc.)	Ignores	Bad (misses features that must interact, "gender" irrelevant given "baby")
Regression Weight (fit least squares, take biggest $\lvert w_j \rvert$)	Bad (can take irrelevant but collinear, can take none of "mom1-3")	Ok (takes "mom" not "grandma", if linear and 'n' large.	Ignores (unless collinear)	Ok (if linear, "gender" relevant give "baby")
Search and Score w/ Validation Error	Ok (takes at least one of "mom" and "mom2")	Bad (takes "grandma", "great-grandma", etc.)	Allows (many false positives)	Ok ("gender" relevant given "baby")
Search and Score w/ L0-norm	Ok (takes exactly one of "mom" and "mom2")	Ok (takes "mom" not grandma if linear-ish).	Ignores (even if collinear)	Ok ("gender" relevant given "baby")

My advice if you want the "relevant" variables.

- Try the association approach.
- Try forward selection with different values of λ.
- Try out a few other feature selection methods too.

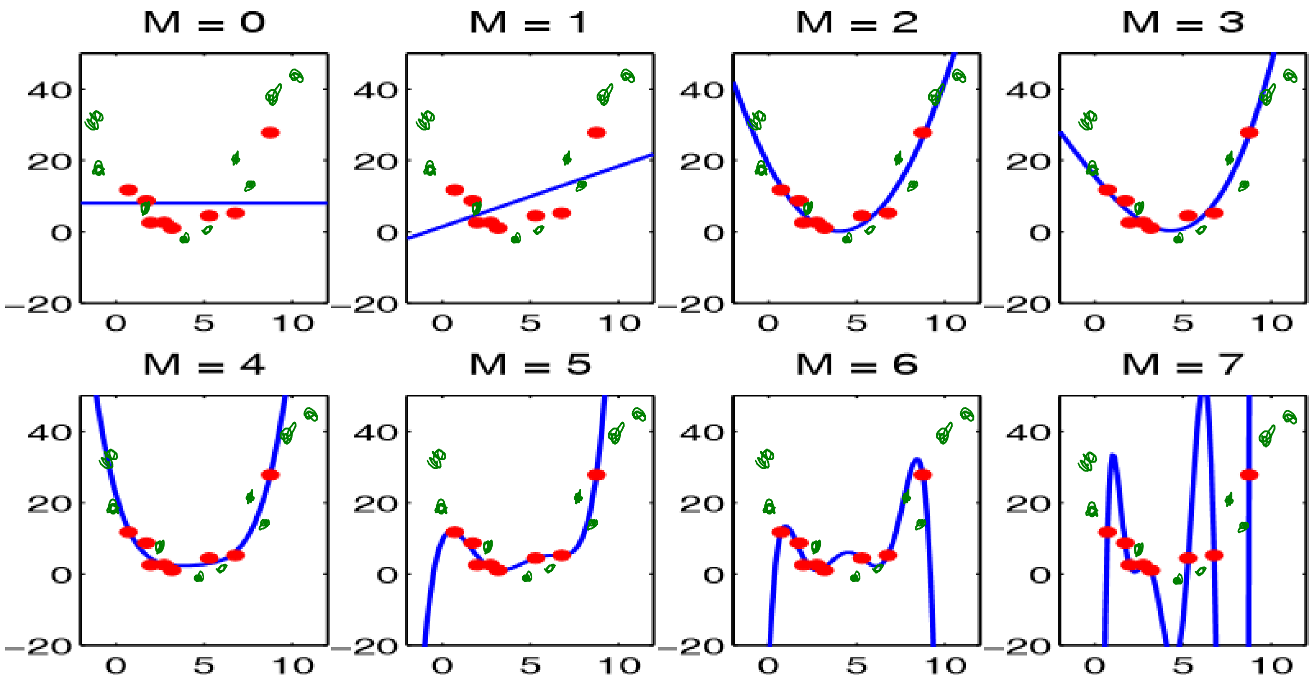
- Discuss the results with the domain expert.
  – They probably have an idea of why some variables might be relevant.

- Don't be overconfident:
  – These methods are probably not discovering how the world truly works.
  – "The model has found that these variables are helpful in predicting y_i."
    - Then a warning that these models are not perfect at finding relevant variables.

(pause)

Recall: Polynomial Degree and Training vs. Testing

- We've said that complicated models tend to overfit more.



- But what if we need a complicated model?

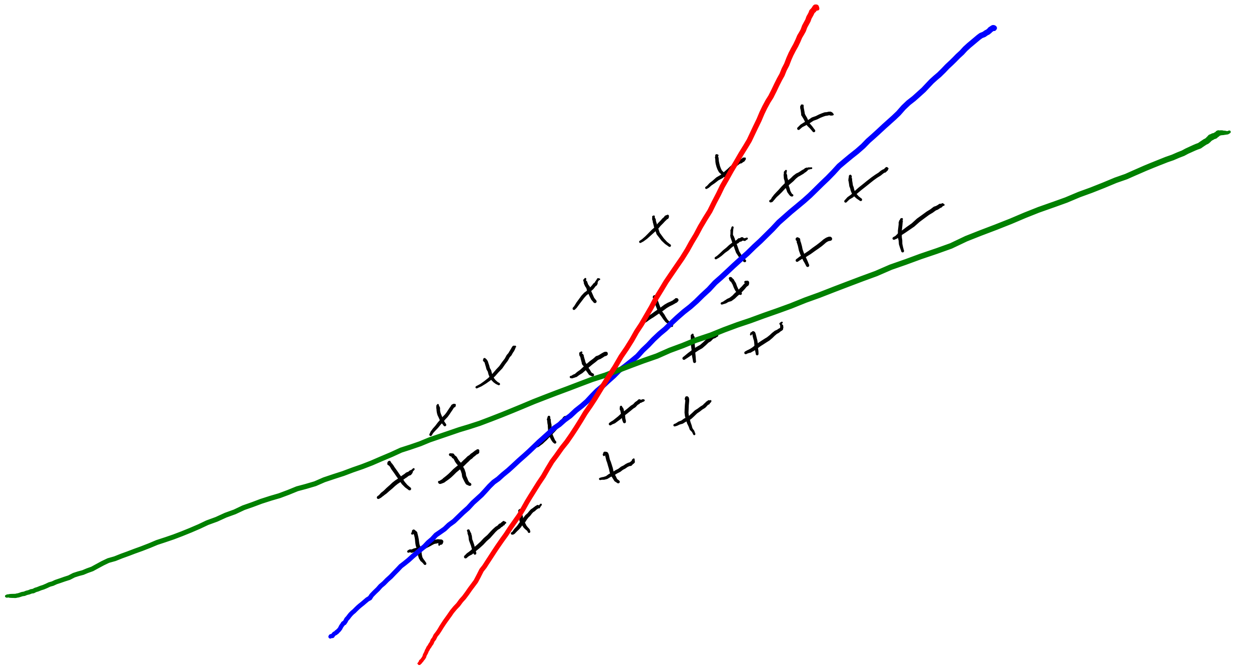
Controlling Complexity

- Usually "true" mapping from x_i to y_i is complex.
  - Might need high-degree polynomial.
  - Might need to combine many features, and don't know "relevant" ones.
- But complex models can overfit.
- So what do we do???

- Our main tools:
  - Model averaging: average over multiple models to decrease variance.
  - Regularization: add a penalty on the complexity of the model.

Would you rather?

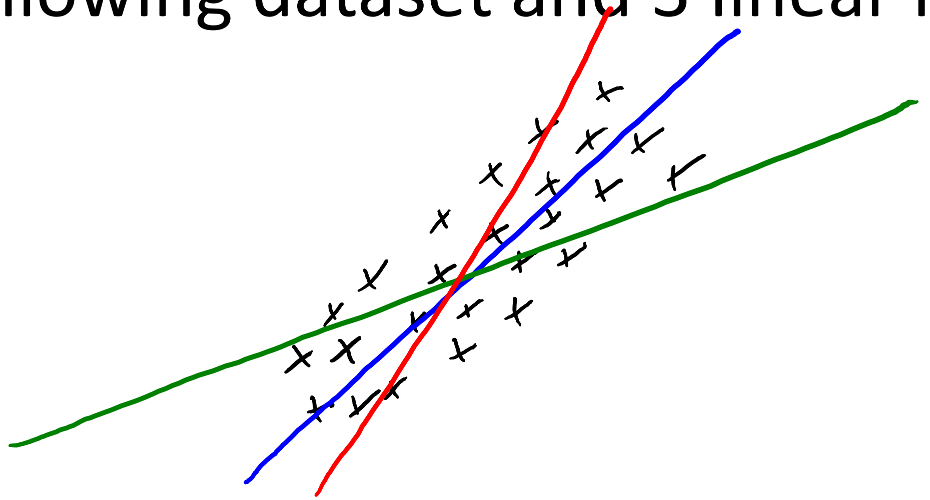
- Consider the following dataset and 3 linear regression models:



- Which line should we choose?

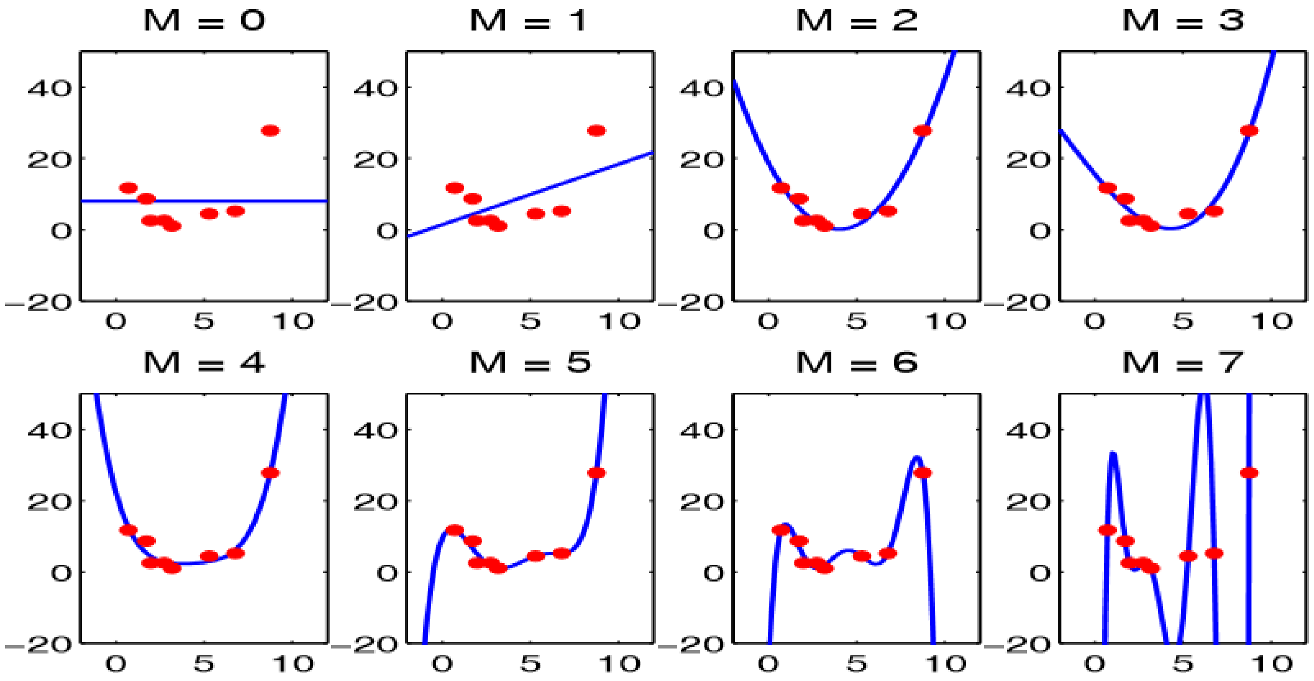
Would you rather?

- Consider the following dataset and 3 linear regression models:



- What if you are forced to choose between red and green?
  - For example, if you used blue with other features it gets a higher error.
- Key idea of regularization:
  - Red line is much more sensitive to this feature, we should pick green.
  - If we don't get the slope right, red line causes more harm than green.

Size of Regression Weights are Overfitting



- The regression weights w_j with degree-7 are huge in this example.
- The degree-7 polynomial would be less sensitive to the data,

  if we "regularized" the w_j so that they are small:

$\hat{y}_i = 0.0001(x_i)^7 + 0.03(x_i)^3 + 3$ vs. $\hat{y}_i = 1000(x_i)^7 - 500(x_i)^6 + 890x_i$

L2-Regularization

- Standard regularization strategy is L2-regularization:

$$f(w) = \frac{1}{2}\sum_{i=1}^{n}(w^T x_i - y_i)^2 + \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2 \quad \text{or} \quad f(w) = \frac{1}{2}\|Xw - y\|^2 + \frac{\lambda}{2}\|w\|^2$$

- Intuition: large slopes w_j tend to lead to overfitting.
- So we minimize squared error plus penalty on L2-norm of 'w'.
  - This objective balances getting low error vs. having small slopes 'w_j'.
    - "You can increase the training error if it makes 'w' much smaller."
    - Nearly-always reduces overfitting.
  - Regularization parameter $\lambda > 0$ controls "strength" of regularization.
    - Large λ puts large penalty on slopes.

L2-Regularization

- Standard regularization strategy is L2-regularization:

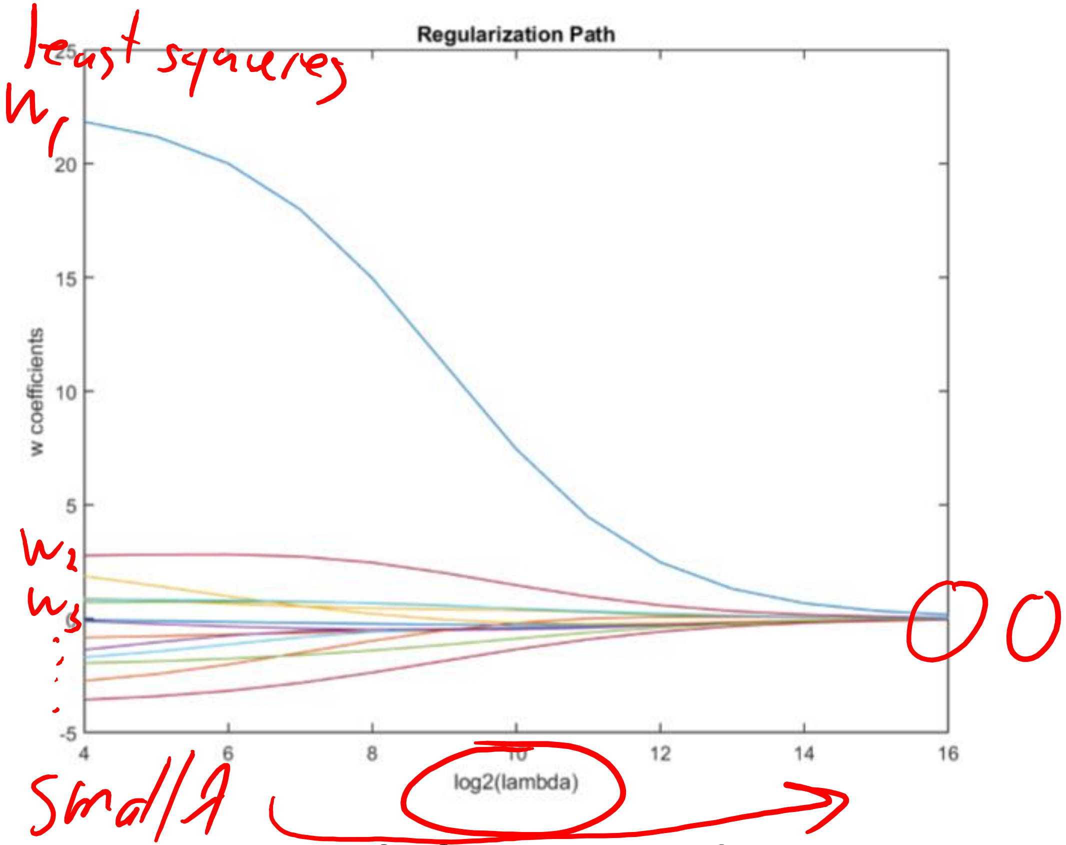
$$f(w) = \frac{1}{2}\sum_{i=1}^{n}(w^Tx_i - y_i)^2 + \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2 \quad \text{or} \quad f(w) = \frac{1}{2}\|Xw - y\|^2 + \frac{\lambda}{2}\|w\|^2$$

- In terms of fundamental trade-off:
  - Regularization increases training error.
  - Regularization decreases approximation error.

- How should you choose λ?
  - Theory: as 'n' grows λ should be in the range O(1) to ($n^{1/2}$).
  - Practice: optimize validation set or cross-validation error.
    - This almost always decreases the test error.

Regularization Path

- Regularization path is a plot of the optimal weights ‘w_j’ as ‘λ’ varies:



- Starts with least squares with λ= 0, and w_j converge to 0 as λ grows.

L2-regularization and the normal equations

- When using L2-regularization we can still set ∇ f(w) to 0 and solve.
- Loss before: $f(w) = ||Xw - y||_2^2$
- Loss after: $f(w) = ||Xw - y||_2^2 + \lambda ||w||_2^2$

- Gradient before: $\nabla f(w) = X^T X w - X^T y$
- Gradient after: $\nabla f(w) = X^T X w - X^T y + \lambda w$

- Linear system before: $X^T X w = X^T y$
- Linear system after: $(X^T X + \lambda I) w = X^T y$
- But unlike $X^T X$, the matrix $(X^T X + \lambda I)$ is always invertible:
  - Multiply by its inverse for unique solution: $w = (X^T X + \lambda I)^{-1} (X^T y)$

Why use L2-Regularization?

- It’s a weird thing to do, but Mark says “always use regularization”.
  - “Almost always decreases test error” should already convince you.

- But here are 6 more reasons:
  1. Solution ‘w’ is unique.
  2. X^TX does not need to be invertible (no collinearity issues).
  3. Less sensitive to changes in X or y.
  4. Gradient descent converge faster (bigger λ means fewer iterations).
  5. Stein’s paradox: if $d \geq 3$, ‘shrinking’ moves us closer to ‘true’ w.
  6. Worst case: just set λ small and get the same performance.

Summary

- "Relevance" is really hard to define.
  - Different methods have different effects on what you find.
- Regularization:
  - Adding a penalty on model complexity.
- L2-regularization: penalty on L2-norm of regression weights 'w'.
  - Almost always improves test error.
  - Simple closed-form unique solution (post-lecture slides).

- Next time: midterm.

Alternative to Search and Score: good old p-values

- Hypothesis testing ("constraint-based") approach:
  - Generalization of the "association" approach to feature selection.
  - Performs a sequence of conditional independence tests.

$x_{ij} \perp y_i \mid x_{is}$

feature 'j'   label   other features 's'

"If I know features in 's' does feature 'j' tell me anything about label?"

  - If they are independent (like "p < .05"), say that 'j' is "irrelevant".
- Common way to do the tests:
  - "Partial" correlation (numerical data).
  - "Conditional" mutual information (discrete data).

Testing-Based Feature Selection

- Hypothesis testing ("constraint-based") approach:
- Two many possible tests, "greedy" method is for each 'j' do:

First test if $x_{ij} \perp y_i$

If still dependent test $x_{ij} \perp y_i \mid x_{is}$ where 's' has one feature

If still dependent test $x_{ij} \perp y_i \mid x_{is}$ where 's' now has two features

Often choose features to minimize dependence.

⋮

If still dependent when 's' includes all other features, declare 'j' relevant.

- "Association approach" is the greedy method where you only do the first test (subsequent tests remove a lot of false positives).

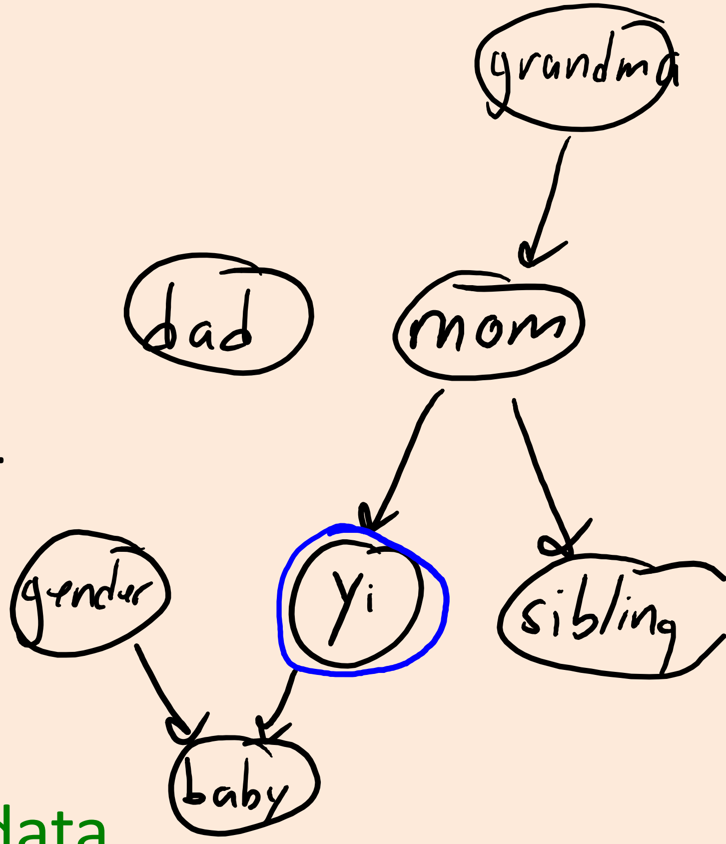
Hypothesis-Based Feature Selection

- Advantages:
  - Deals with conditional independence.
  - Algorithm can explain why it thinks ‘j’ is irrelevant.
  - Doesn’t necessarily need linearity.
- Disadvantages:
  - Deals badly with exact dependence: doesn’t select “mom” or “mom2” if both present.
  - Usual warning about testing multiple hypotheses:
    - If you test $p < 0.05$ more than 20 times, you’re going to make errors.
  - Greedy approach may be sub-optimal.
- Neither good nor bad:
  - Allows tiny effects.
  - Says “gender” is irrelevant when you know “baby”.
  - This approach is sometimes better for finding relevant factors, not to select features for learning.

Causality

- None of these approaches address causality or confounding:
  - "Mom" is the only relevant direct causal factor.
  - "Dad" is really irrelevant.
  - "Grandma" is causal but is irrelevant if we know "mom".

- Other factors can help prediction but aren't causal:
  - "Sibling" is predictive due to confounding of effect of same "mom".
  - "Baby" is predictive due to reverse causality.
  - "Gender" is predictive due to common effect on "baby".



- We can sometimes address this using interventional data…

Interventional Data

- The difference between observational and interventional data:
  - If I see that my watch says 10:45, class is almost over (observational).
  - If I set my watch to say 10:45, it doesn't help (interventional).
- The intervention can help discover causal effects:
  - "Watch" is only predictive of "time" in observational setting (so not causal).
- General idea for identifying causal effects:
  - "Force" the variable to take a certain value, then measure the effect.
    - If the dependency remains, there is a causal effect.
    - We "break" connections from reverse causality, common effects, or confounding.

Causality and Dataset Collection

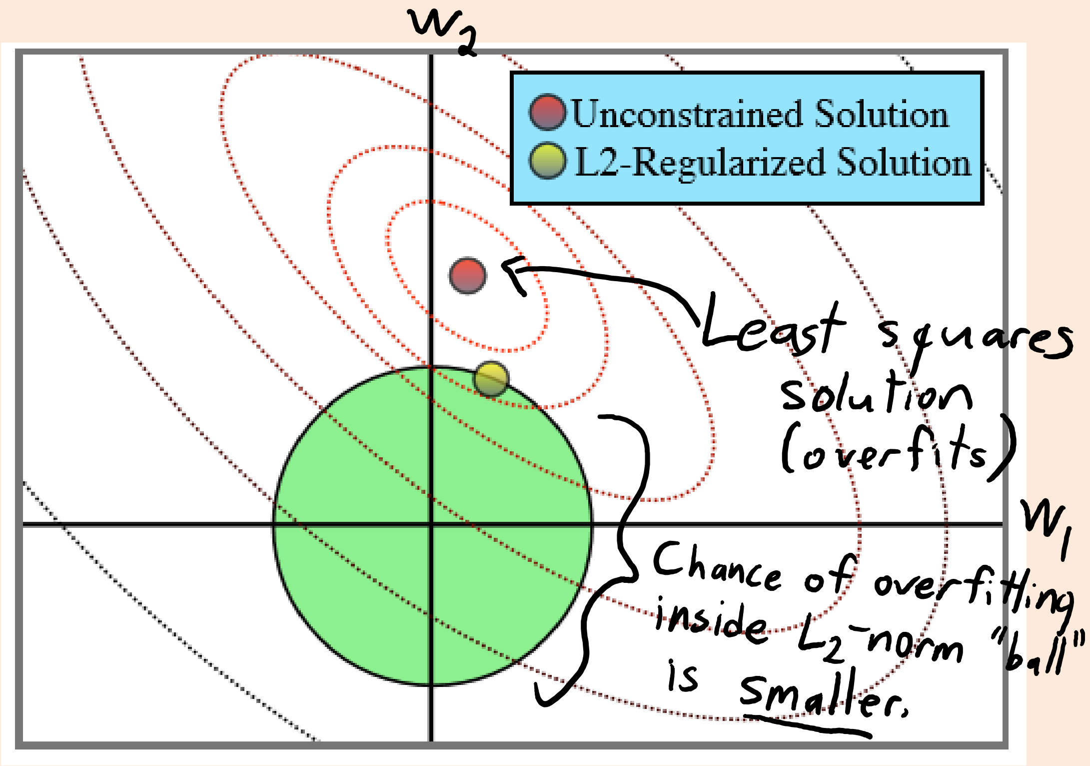
- This has to do with the way you collect data:
  - You can’t “look” for variables taking the value “after the fact”.
  - You need to manipulate the value of the variable, then watch for changes.
- This is the basis for randomized control trial in medicine:
  - Randomly assigning pills “forces” value of “treatment” variable.
  - Include a “control” as a value to prevent placebo effect as confounding.
- See also Simpson’s Paradox:
  - https://www.youtube.com/watch?v=ebEkn-BiW5k

L2-Regularization

- Standard regularization strategy is L2-regularization:

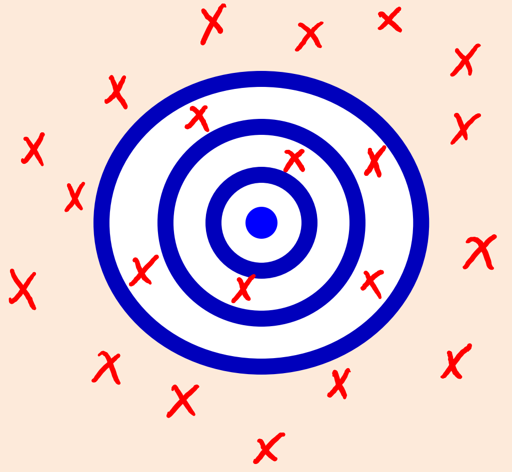
$$f(w) = \frac{1}{2}\sum_{i=1}^{n}(w^Tx_i - y_i)^2 + \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2 \quad \text{or} \quad f(w) = \frac{1}{2}\|Xw - y\|^2 + \frac{\lambda}{2}\|w\|^2$$

- Equivalent to minimizing squared error but keeping L2-norm small.



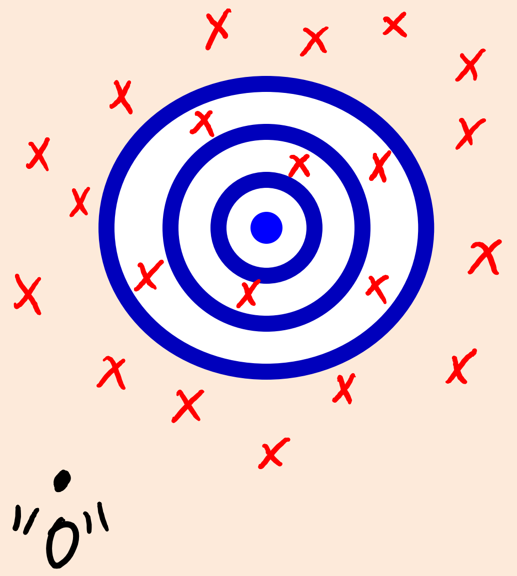
Regularization/Shrinking Paradox

- We throw darts at a target:
  - Assume we don’t always hit the exact center.
  - Assume the darts follow a symmetric pattern around center.



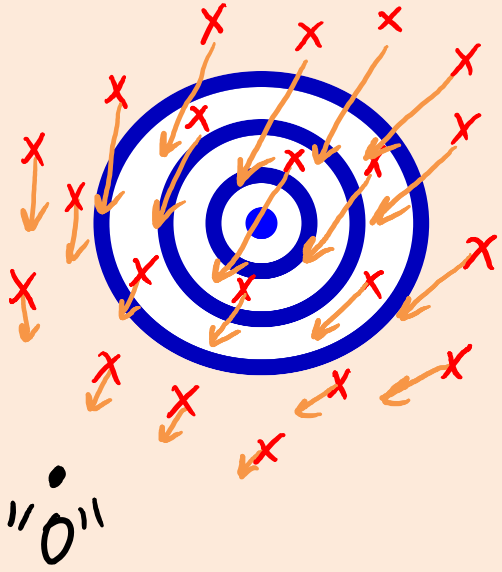
Regularization/Shrinking Paradox

- We throw darts at a target:
  - Assume we don’t always hit the exact center.
  - Assume the darts follow a symmetric pattern around center.
- Shrinkage of the darts :
  1. Choose some arbitrary location ‘0’.
  2. Measure distances from darts to ‘0’.



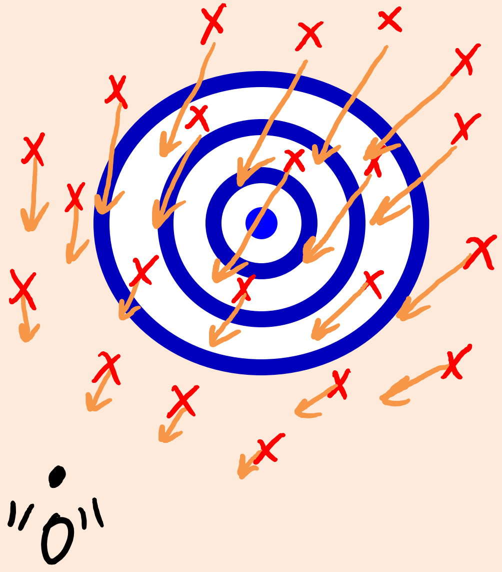
Regularization/Shrinking Paradox

- We throw darts at a target:
  - Assume we don't always hit the exact center.
  - Assume the darts follow a symmetric pattern around center.
- Shrinkage of the darts :
  1. Choose some arbitrary location '0'.
  2. Measure distances from darts to '0'.
  3. Move misses towards '0', by *small* amount proportional to distance from 0.
- If small enough, darts will be closer to center on average.



Regularization/Shrinking Paradox

- We throw darts at a target:
  - Assume we don't always hit the exact center.
  - Assume the darts follow a symmetric pattern around center.
- Shrinkage of the darts :
  1. Choose some arbitrary location '0'.
  2. Measure distances from darts to '0'.
  3. Move misses towards '0', by *small* amount proportional to distance from 0.
- If small enough, darts will be closer to center on average.



Visualization of the related higher-dimensional paradox that the mean of data coming from a Gaussian is not the best estimate of the mean of the Gaussian in 3-dimensions or higher: https://www.naftaliharris.com/blog/steinviz