CPSC 340: Machine Learning and Data Mining

Regularization
Fall 2015
Admin

• No tutorials/class Monday (holiday).
Radial Basis Functions

- Alternative to polynomial bases are radial basis functions (RBFs):
  - Basis functions that depend on distances to training points.
  - A non-parametric basis.
- Most common example is Gaussian RBF:
  \[ k(x_i, x_j) = \frac{1}{\sigma^2} \exp\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right) \]
  \[ X_{rbf} = \begin{pmatrix}
  k(x_1, x_1) & k(x_1, x_2) & \cdots & k(x_1, x_n) \\
  k(x_2, x_1) & k(x_2, x_2) & \cdots & k(x_2, x_n) \\
  \vdots & \vdots & \ddots & \vdots \\
  k(x_n, x_1) & k(x_n, x_2) & \cdots & k(x_n, x_n)
  \end{pmatrix} \]
- Variance \(\sigma^2\) controls how much nearby vs. far points contribute.
  - Affects fundamental trade-off.
- There are universal consistency results with these functions:
  - In terms of bias-variance, achieves irreducible error as ‘n’ goes to infinity.
Predicting the Future

• In principle, we can use any features $x_i$ that we think are relevant.
• This makes it tempting to use time as a feature, and predict future.
Predicting the Future

• In principle, we can use any features $x_i$ that we think are relevant.
• This makes it tempting to use \textit{time} as a feature, and predict future.

We need to be cautious about doing this.
Predicting 100m times 500 years in the future?
Predicting 100m times 400 years in the future?

Limit is 9.48 seconds, reached in 500 years.

https://plus.maths.org/content/sites/plus.maths.org/files/articles/2011/usain/graph2.gif
No Free Lunch, Consistency, and the Future
No Free Lunch, Consistency, and the Future

[Hand-drawn graph]

- Least squares seems like a good fit.
- Time
- First available measurement
- Present
No Free Lunch, Consistency, and the Future

this model also fits data well.

First available measurement.

On the other hand, training error is likely to approximate test error in this model.

Least squares seems like a good fit.
Ockham’s Razor vs. No Free Lunch

- **Ockham’s razor** is a problem-solving principle:
  - “Among competing hypotheses, the one with the fewest assumptions should be selected.”
  - Suggests we should select linear model.

- **Fundamental theorem of ML:**
  - If training same error, pick model less likely to overfit.
  - Formal version of Occam’s problem-solving principle.
  - Also suggests we should select linear model.

- **No free lunch theorem:**
  - There *exists possible datasets* where you should select the green model.
No Free Lunch, Consistency, and the Future

Let's Collect more data.

first available measurement.  

Time  

present.
No Free Lunch, Consistency, and the Future

New data agrees with green model, seems more plausible.

But there is still no free lunch!
No Free Lunch, Consistency, and the Future

Collect even more data.

↑ present.
No Free Lunch, Consistency, and the Future

New data shows that green model is not perfect.

But with universally consistent estimator, we can use more data to get closer to true model.
No Free Lunch, Consistency, and the Future

Converge to best model as $n \to \infty$, if we use a "universally consistent" method.
No Free Lunch, Consistency, and the Future

We don't get data from the future. No matter how much data you have in the present, without assumptions it says nothing about future.

"Consistency zone."

"There really is no free lunch zone."
No Free Lunch, Consistency, and the Future
Application: Climate Models

• Has Earth warmed up over last 100 years? (Consistency zone)
  – Data clearly says ‘yes’.

• Will Earth continue to warm over next 100 years? (Really NFL zone)
  – We should be more skeptical about models that predict future events.
Application: Climate Models

- So should we all become global warming skeptics?
- If we average over models that overfit in *different* ways, we expect the test error to be lower, so this gives more confidence:
  - We should be skeptical of individual models, but agreeing predictions made by models with different data/assumptions are more likely be true.
- If all near-future predictions agree, they are likely to be accurate.
- As we go further in the future, variance of average will be higher.
Regularization

• **Ridge regression** is a very common variation on least squares:

\[
\arg\min_{w \in \mathbb{R}^d} \frac{1}{2} \sum_{i=1}^{n} (y_i - w^T x_i)^2 + \frac{\lambda}{2} \sum_{j=1}^{d} w_j^2
\]

• The extra term is called **L2-regularization**:
  – Objective balances getting low error vs. having small slope ‘w’.
  – E.g., you allow a small increase in error if it makes slope ‘w’ much smaller.

• **Regularization parameter** \( \lambda > 0 \) controls level of regularization:
  – High \( \lambda \) makes L2-norm more important compared to than data.
  – Theory says choices should be in the range \( O(1) \) to \( O(n^{-1/2}) \).
  – In practice, set by validation set or cross-validation.
Why use L2-Regularization?

• It’s a weird thing to do, but **L2-regularization is magic.**

• 6 reasons to use L2-regularization:
  1. Does not require $X'X$ to be invertible.
  2. Solution ‘$w$’ is unique.
  3. Solution ‘$w$’ is less sensitive to changes in $X$ or $y$ (like ensemble methods).
  5. Significant decrease in variance, and often only small increase in bias.
     • This means you typically have **lower test error.**
  6. Stein’s paradox: if $d \geq 3$, ‘shrinking’ estimate moves us closer to ‘true’ $w$. 

---

**Note:** The annotations on the image suggest that the question was about how well the model performs in terms of train error and test error. The text indicates that while L2-regularization decreases variance, it often increases bias slightly. This trade-off affects the model’s performance on both training and test sets.
Shrinking is Weird and Magical

• We throw darts at a target:
  – Assume we don’t always hit the exact center.
  – Assume the darts follow a symmetric pattern around center.
Shrinking is Weird and Magical

• 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’.
Shrinking is Weird and Magical

• 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 distances.

• On average, darts will be closer to center.
Ridge Regression Calculation

Objective: \[ f(w) = \frac{1}{2}(y - Xw)^T(y - Xw) + \frac{\lambda}{2} w^T w. \]

Gradient: \[ \nabla f(w) = X^T X w - X^T y + \lambda w \]

Setting to zero: \[ X^T X w + \lambda w = X^T y \] or \[ (X^T X + \lambda I)w = X^T y. \]

Pre-multiply by \( (X^T X + \lambda I)^{-1} \), which always exists:

\[ w = (X^T X + \lambda I)^{-1} X^T y. \]
Least Squares with Outliers

- Consider least squares problem with outliers:

\[ x \leftarrow \text{"outlier"}; \text{it's not like the others.} \]

This is probably what we want.
Least Squares with Outliers

• Consider least squares problem with outliers:

\[ x \leftarrow \text{"outlier"}; \text{it's not like the others.} \]

• Least squares is very sensitive to outliers.
Least Squares with Outliers

- Squaring error shrinks small errors, and magnifies large errors:

\[
\begin{align*}
\text{Absolute errors} & \quad \text{Square errors} \\
(5)^2 &= 25 \\
\left(\frac{1}{2}\right)^2 &= \frac{1}{4}
\end{align*}
\]

- Outliers (large error) influence ‘w’ much more than other points.
Least Squares with Outliers

• Squaring error shrinks small errors, and magnifies large errors:

- Outliers (large error) influence ‘w’ much more than other points.
  - Good if outlier means ‘plane crashes’, bad if it means ‘data entry error’.
Robust Regression

• Robust regression objectives put less focus on far-away points.
• For example, use absolute error:
\[
\arg \min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} \left| y_i - w^T x_i \right|
\]
• Now decreasing ‘small’ and ‘large’ errors is equally important.
• Norms are a nice way to write least squares vs. least absolute error:

Define ‘residual’ vector \( \mathbf{r} \) with elements \( r_i = (y_i - w^T x_i) \).

Least squares:
\[
\sum_{i=1}^{n} (y_i - w^T x_i)^2 = \sum_{i=1}^{n} r_i^2 \\
= \| \mathbf{r} \|_2^2 \quad \Rightarrow \text{\( L_2 \)-norm of residuals}
\]

Least absolute error:
\[
\sum_{i=1}^{n} |y_i - w^T x_i| = \sum_{i=1}^{n} |r_i| \\
= \| \mathbf{r} \|_1 \quad \Rightarrow \text{\( L_1 \)-norm of residuals}
\]
Summary

• **Predicting future is hard**, ensemble predictions are more reliable.
• **Regularization** improves test error because it is magic.
• **Outliers** can cause least squares to perform poorly.
• **Robust regression** using L1-norm is less sensitive.

• Next time:
  – How to fine the L1-norm solution, and what if features are irrelevant?