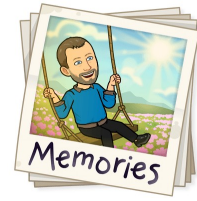


Machine Learning

Textbook: Machine Learning Refined, by Watt, Borhani, and Katsaggelos, 2nd edition:
MLRefined.com

Previous Lecture

- ♦ The design matrix X stores all observations and all linear features.
- ♦ All predictions are produced at once by $\hat{\mathbf{y}} = X\mathbf{w}$.
- ♦ Least squares minimizes $\|X\mathbf{w} - \mathbf{y}\|^2$.
- ♦ Setting the gradient to zero gives the normal equations $X^T X \mathbf{w} = X^T \mathbf{y}$.
- ♦ Geometrically, $X\mathbf{w}^*$ is the orthogonal projection of $\mathbf{y}$ onto the column space of X .
- ♦ Full column rank gives a unique least-squares parameter vector;
dependent columns create nonuniqueness.



§5.3 - Alternatives to Least Squares

Last time:

least squares minimized

$$\frac{1}{P} \sum_{p=1}^P r_p^2.$$

Today:

what changes if we measure error differently?

In particular:

squared error versus absolute error.

The Choice of Loss Matters

A residual is

$$r_p = \hat{y}_p - y_p.$$

Least squares uses the penalty

$$r_p^2.$$

Another natural choice is

$$|r_p|.$$

Both are zero for a perfect prediction.

But they treat large errors very differently.

L^2 Versus L^1

For residual vector

$$\mathbf{r} = \begin{bmatrix} r_1 \\ \vdots \\ r_P \end{bmatrix},$$

least squares minimizes

$$g_2(\mathbf{w}) = \frac{1}{P} \|\mathbf{r}\|_2^2 = \frac{1}{P} \sum_{p=1}^P r_p^2.$$

Least absolute deviations minimizes

$$g_1(\mathbf{w}) = \frac{1}{P} \|\mathbf{r}\|_1 = \frac{1}{P} \sum_{p=1}^P |r_p|.$$

This is called **least absolute deviations**,
or **LAD regression**.

Compare the Penalties

For one residual:

$ r $	1	2	5	10
$ r $	1	2	5	10
r^2	1	4	25	100

Doubling a residual:

absolute penalty doubles,
squared penalty quadruples.

Large residuals have much more influence under squared error.

The Derivatives Tell the Same Story

For squared loss

$$\ell_2(r) = r^2,$$

we have

$$\ell_2'(r) = 2r.$$

Magnitude of slope:

$$2|r|.$$

Large residual:

large pull on the optimization.

For absolute loss

$$\ell_1(r) = |r|,$$

away from $r = 0$:

$$\ell_1'(r) = \begin{cases} -1, & r < 0, \\ 1, & r > 0. \end{cases}$$

Magnitude of slope:
1.

Large residuals do not receive an ever-increasing slope.

But $|r|$ Has a Corner

At

$$r = 0,$$

$|r|$ is not differentiable.

So the LAD cost can have corners.

Consequences:

- ordinary gradient formulas may fail at some points;

- Newton's method cannot be applied in the same simple way;

- derivative-free methods remain possible.

Later optimization tools can also handle nonsmooth objectives.

Smoothness was one important advantage of least squares.

A Very Simple Regression Model

Forget the inputs temporarily.

Suppose our model predicts one constant:

$$\hat{y} = w.$$

Least squares chooses w to minimize

$$g_2(w) = \frac{1}{P} \sum_{p=1}^P (w - y_p)^2.$$

Differentiate:

$$g_2'(w) = \frac{2}{P} \sum_{p=1}^P (w - y_p).$$

Set equal to zero:

$$Pw - \sum y_p = 0.$$

Therefore

$$w_{L^2}^* = \frac{1}{P} \sum_{p=1}^P y_p = \bar{y}.$$

Least squares chooses the **mean**.

What Does L^1 Choose?

Now minimize

$$g_1(w) = \frac{1}{P} \sum_{p=1}^P |w - y_p|.$$

As w moves to the right:
observations to the left pull the slope upward;

observations to the right pull the slope downward.

The balance occurs at a median.

Thus
 $w_{L^1}^*$ = a median of the data.

Least squares $\rightarrow$ mean.

Least absolute deviations $\rightarrow$ median.

An Outlier Example

Suppose
 $y = 1, 2, 2, 3, 20$.

Least-squares constant:
 $\bar{y} = \frac{1+2+2+3+20}{5} = 5.6$.

LAD constant:
 $\text{median}(y) = 2$.

One large observation pulls the mean far upward.

The median does not move.

Compare With and Without the Outlier

Without the outlier, use
 $1, 2, 2, 3, 4$.

Mean:
 2.4 .

Median:
 2 .

Replace 4 by 20.

New mean:
 5.6 .

New median:
 2 .

The mean changes by
 3.2 .

The median changes by
0.

This is the basic robustness advantage of L^1 loss.

Each Loss Wins Its Own Contest

For the data

1, 2, 2, 3, 20:

At the L^2 solution $w = 5.6$:

$$g_2(5.6) = 52.24,$$

$$g_1(5.6) = 5.76.$$

At the L^1 solution $w = 2$:

$$g_2(2) = 65.2,$$

$$g_1(2) = 4.0.$$

So:

$w = 5.6$ is better under squared error;

$w = 2$ is better under absolute error.

There is no contradiction.

They optimize different objectives.

Back to a Regression Line

Consider

x	0	1	2	3	4
y	0	1	2	3	20

Four points lie exactly on

$$y = x.$$

The final point is a large vertical outlier.

What will the two losses do?

Least Squares Compromises With the Outlier

Least-squares fit:

$$\hat{y} = -3.2 + 4.2x.$$

Residuals:

$$-3.2, 0, 3.2, 6.4, -6.4.$$

MSE:
20.48.

MAE:
3.84.

The outlier pulls the entire fitted line away from
 $y = x$.

Least Absolute Deviations

For the same data, an LAD fit is
 $\hat{y} = x$.

Residuals:
0, 0, 0, 0, -16.

MSE:
51.2.

MAE:
3.2.

The LAD fit accepts one large residual
rather than moving away from four perfectly aligned observations.

Again:
least squares has smaller MSE;

LAD has smaller MAE.

Each minimizes its own loss.

Robust Does Not Mean Immune

LAD is less sensitive to large **vertical** residuals.

But a point with an extreme input value can still have strong influence.

For example:
a point very far away in x
can strongly affect the orientation of a fitted line.

So:
LAD is more robust to some outliers,

but it does not make regression immune to unusual observations.

The Geometry Changes

Least squares:

$$\|\mathbf{r}\|_2^2.$$

Level sets in residual space:
circles or spheres.

LAD:
$$\|\mathbf{r}\|_1.$$

Level sets:
diamonds in two dimensions,
cross-polytopes in higher dimensions.

Corners in the L^1 geometry
correspond to nonsmoothness of the absolute-value loss.

The choice of norm changes both
geometry and optimization.

No Normal Equations for LAD

Least squares:
$$g_2(\mathbf{w}) = \frac{1}{P} \|X\mathbf{w} - \mathbf{y}\|_2^2.$$

Differentiating gives
$$X^T X \mathbf{w} = X^T \mathbf{y}.$$

LAD:
$$g_1(\mathbf{w}) = \frac{1}{P} \|X\mathbf{w} - \mathbf{y}\|_1.$$

Because of the absolute values:
no analogous single smooth normal-equation formula.

We have traded algebraic convenience
for greater robustness to large residuals.

Absolute Error in Python

For a set of predictions:

```
def mae(y, y_pred):  
    return np.mean(np.abs(y_pred - y))
```

New Python idea:
`np`
takes absolute values element by element.

Compare with MSE:

```
def mse(y, y_pred):  
    return np.mean((y_pred - y)**2)
```

Same residuals.

Different penalty.

Mean and Median in Python

For the outlier data:

```
y = np.array([1., 2., 2., 3., 20.])

np.mean(y)
np.median(y)
```

Outputs:

```
5.6
2.0
```

New Python idea:

np
computes the sample median.

Least squares and LAD have selected
two familiar summary statistics.

Define an LAD Regression Cost

Use

```
x = np.array([0., 1., 2., 3., 4.])
y = np.array([0., 1., 2., 3., 20.])

def model(x, w):
    return w[0] + w[1]*x

def lad_cost(w):
    residuals = model(x, w) - y
    return np.mean(np.abs(residuals))
```

Evaluate $y = x$:

```
lad_cost(np.array([0.0, 1.0]))
```

Output:

```
3.2.
```

Reuse a Zero-Order Method

Because LAD is nonsmooth,
we can reuse ideas from Lesson 1.3.

A simple grid search:

```
best_cost = np.inf
best_w = None

for w0 in np.linspace(-2, 2, 81):
    for w1 in np.linspace(0, 2, 81):
        w = np.array([w0, w1])
        c = lad_cost(w)

        if c < best_cost:
            best_cost = c
            best_w = w.copy()
```

For this grid:
best_w = [0, 1].

New Python ideas:
np represents infinity;
None marks “no value yet”;

nested loops search two parameters.

Which Loss Should We Use?

Squared error is attractive when:
large errors should receive especially large penalties;

smooth optimization is important;

the mean is the appropriate notion of center.

Absolute error is attractive when:
robustness to large vertical residuals matters;

the median is the appropriate notion of center.

The loss function encodes what we mean by
“good prediction.”

What did we learn?

- ◆ Least squares minimizes the squared L^2 residual size; LAD minimizes the L^1 residual size.
- ◆ Squared loss gives increasingly large influence to large residuals; absolute loss grows only linearly.
- ◆ For a constant model, least squares chooses the mean while LAD chooses a median.
- ◆ LAD is generally more robust to large vertical outliers than least squares.
- ◆ Absolute loss is nonsmooth at zero, so the normal-equation and ordinary Newton machinery no longer applies directly.
- ◆ Choosing a loss function is part of choosing what “best fit” means.



Next

- A fitted model needs more than an optimization objective.
- We also need ways to describe how well the model explains and predicts the data.
- Next: regression error and goodness-of-fit measures.
- **1.12 - Assessing a Regression Model**