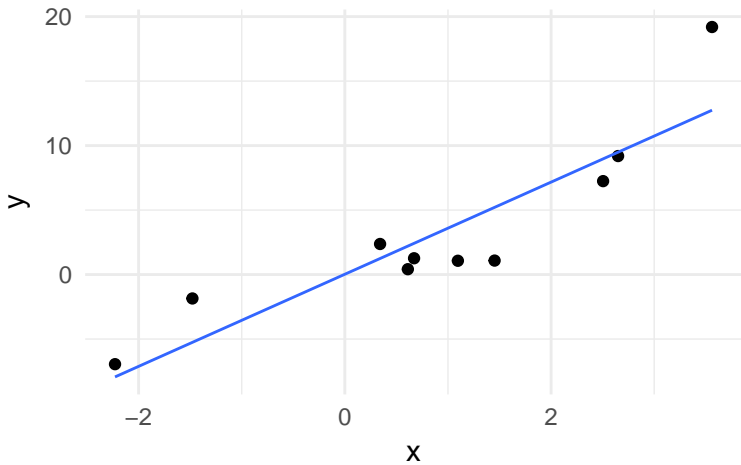


# The problem with parameters: using information theory to evaluate models

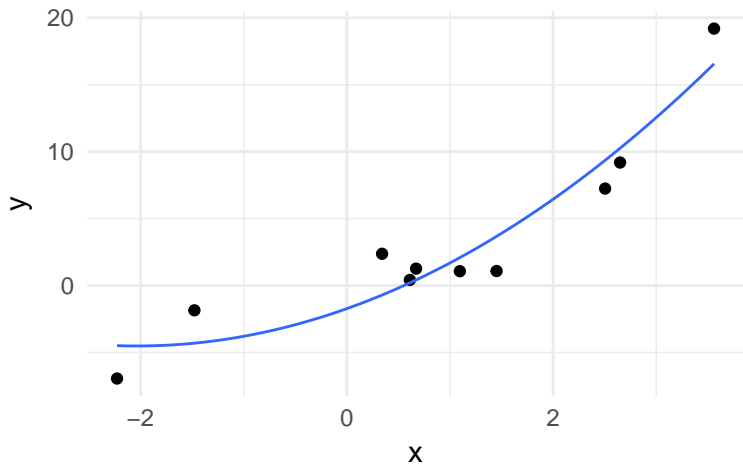
Max Joseph

March 16, 2017

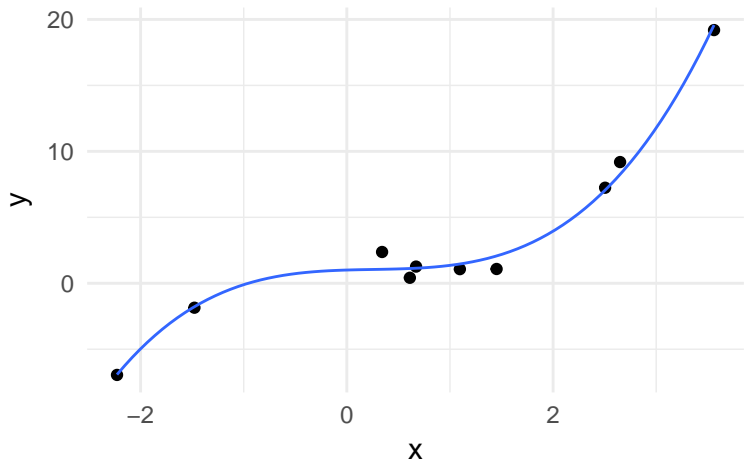
$$\hat{y} = \beta_0 + \beta_1 x$$



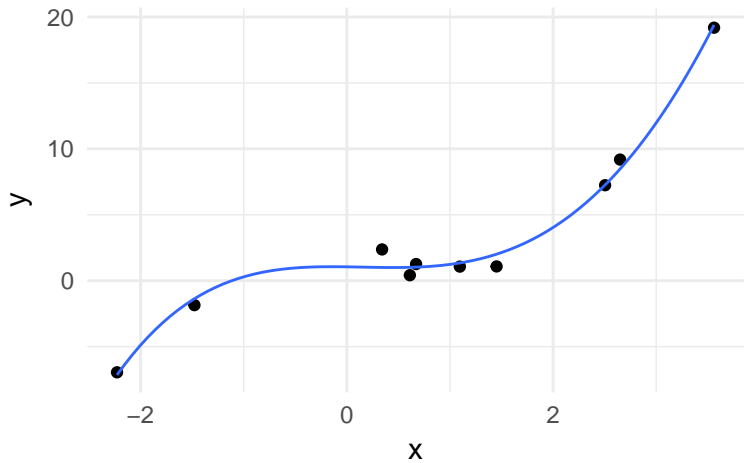
$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2$$



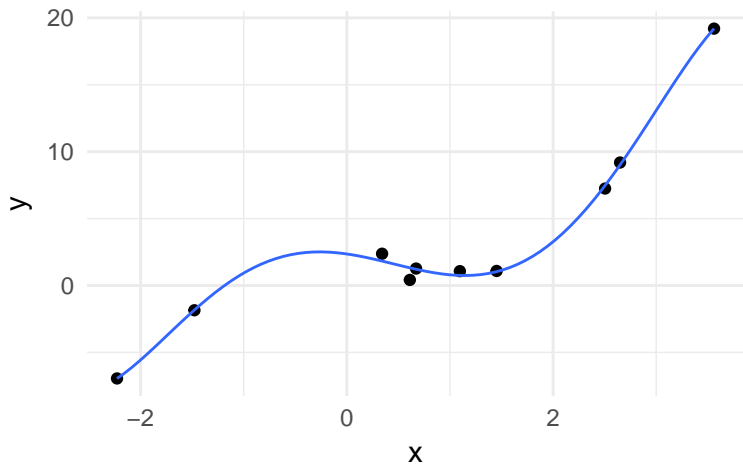
$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$$



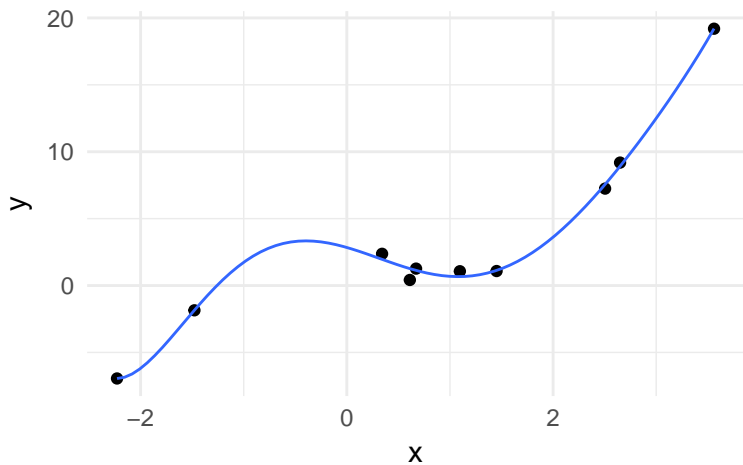
$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4$$



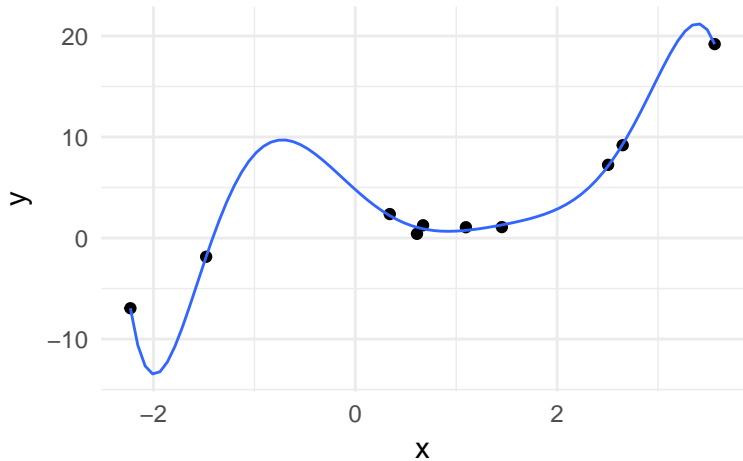
$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5$$



$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5 + \beta_6 x^6$$

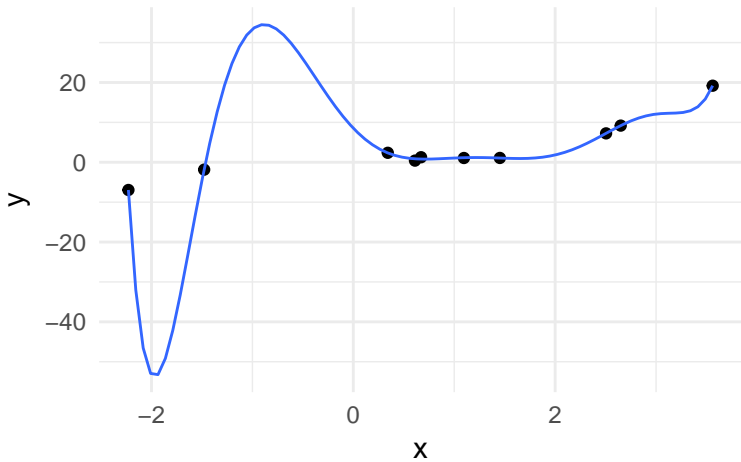


$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5 + \beta_6 x^6 + \beta_7 x^7$$

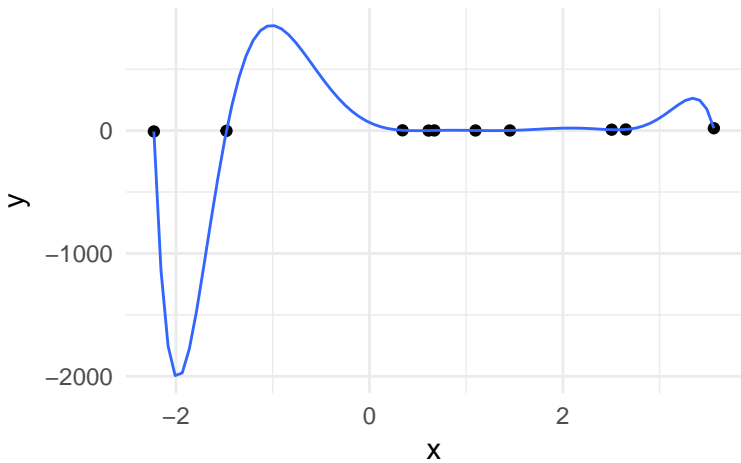




$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5 + \beta_6 x^6 + \beta_7 x^7 + \beta_8 x^8$$



$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5 + \beta_6 x^6 + \beta_7 x^7 + \beta_8 x^8 + \beta_9 x^9$$

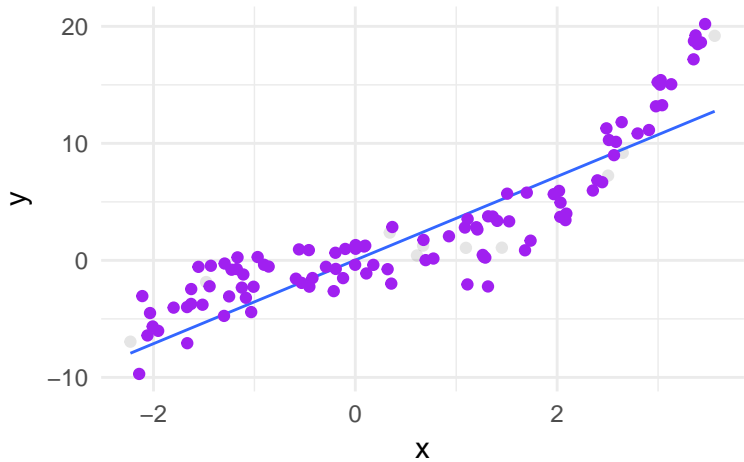


# Overfitting

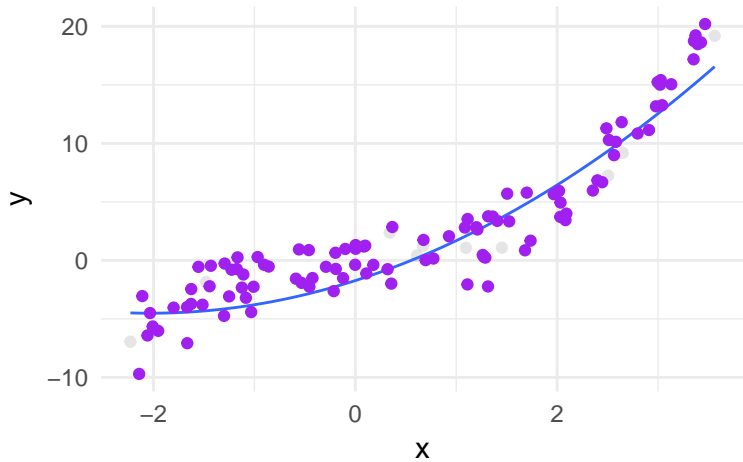
Parameters begin to fit to **noise**

- ▶ good fit to **training data**
- ▶ bad predictions for out of sample data

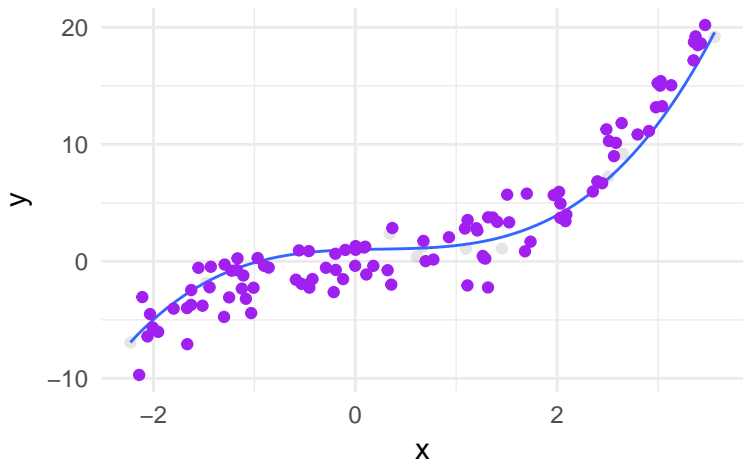
$$\hat{y} = \beta_0 + \beta_1 x$$



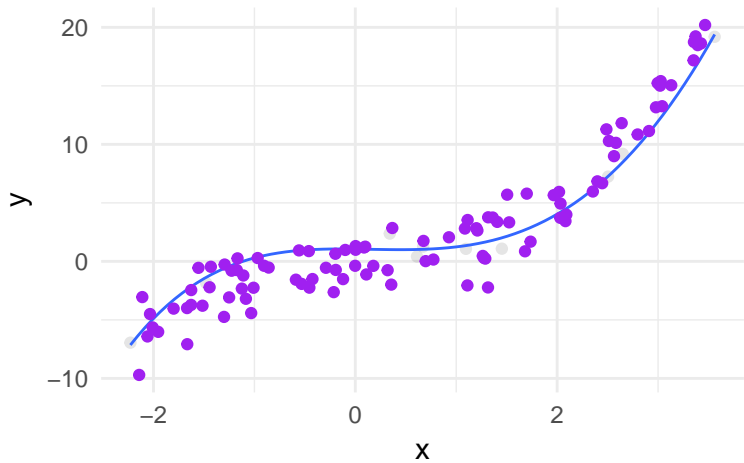
$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2$$



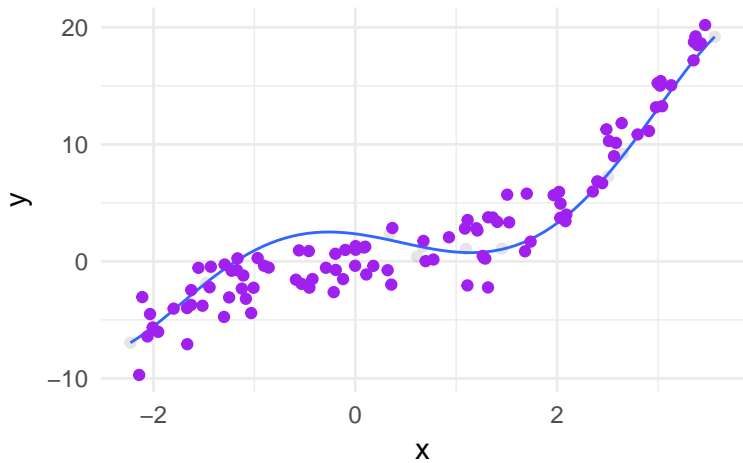
$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$$



$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4$$

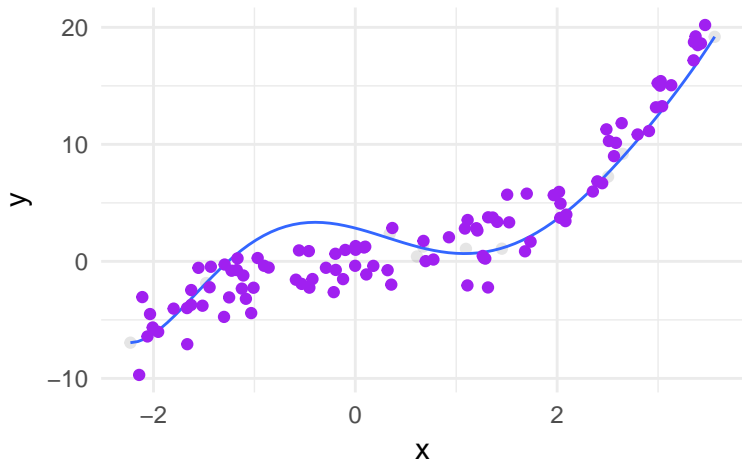


$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5$$

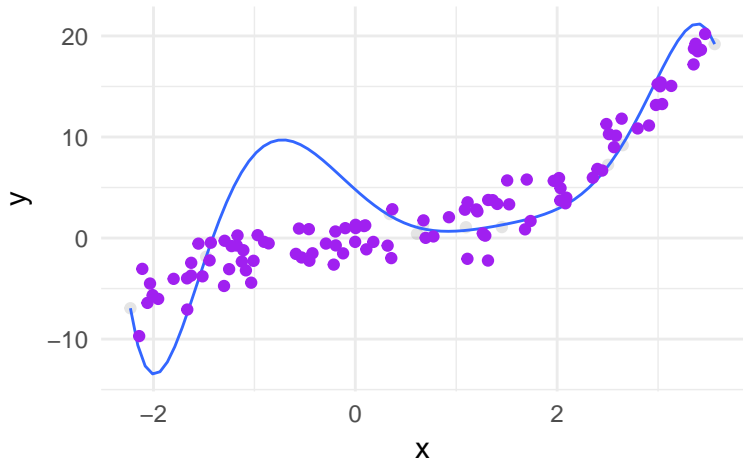




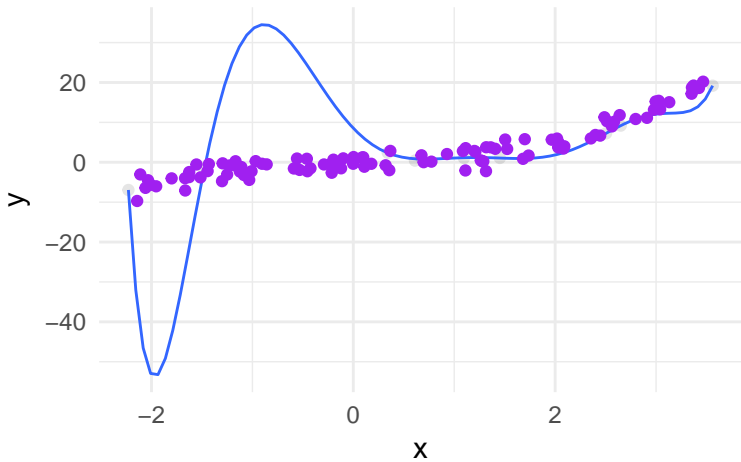
$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5 + \beta_6 x^6$$



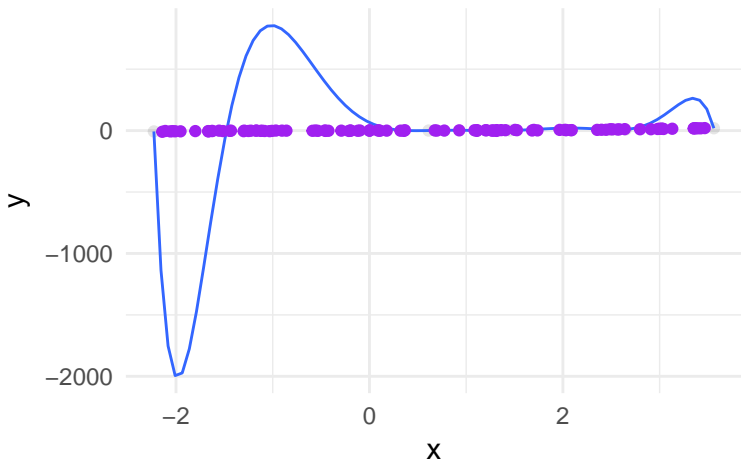
$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5 + \beta_6 x^6 + \beta_7 x^7$$



$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5 + \beta_6 x^6 + \beta_7 x^7 + \beta_8 x^8$$

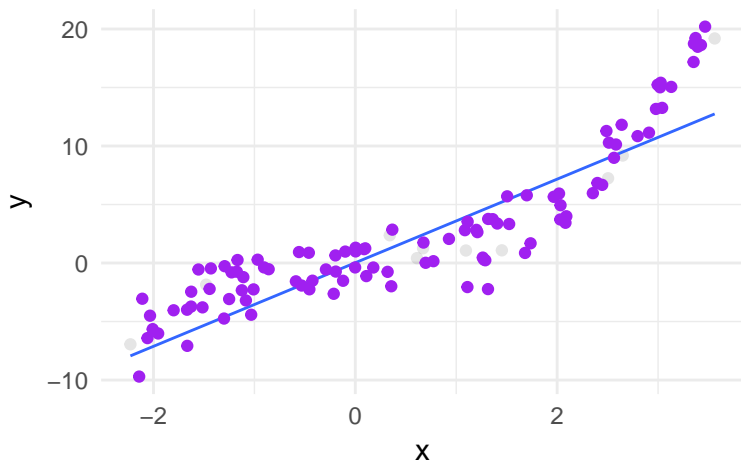


$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5 + \beta_6 x^6 + \beta_7 x^7 + \beta_8 x^8 + \beta_9 x^9$$



# Underfitting

Model is too simplistic to capture signal



# Today

Working toward **information criteria** to balance:

- ▶ model complexity
- ▶ out of sample predictive power

# Roadmap

## **Finding a way to compare models**

1. Information entropy
2. Kullback-Leibler divergence
3. Deviance
4. Akaike's information criterion

# Information entropy

Uncertainty contained in a probability distribution

$$H(p) = - \sum_{i=1}^n p_i \log(p_i)$$

- ▶  $H(p)$ : information entropy of a distribution  $p$
- ▶  $n$ : the number of possible outcomes
- ▶  $p_i$ : the probability of outcome  $i$



## Activity

Compute the information entropy for your die!

$$H(p) = - \sum_{i=1}^n p_i \log(p_i)$$

# What if we didn't know anything about dice?

Find an estimate of information entropy:

1. Estimate  $p_1, p_2, p_3, \dots$
2. Compute information entropy of your estimated distribution:

$$H(\hat{p}) = - \sum_{i=1}^n \hat{p}_i \log(\hat{p}_i)$$

# Divergence

How far off is our model from the true distribution?

## Example

We used  $\hat{p}$  to estimate  $p$

- ▶ What's the “divergence” between  $\hat{p}$  and  $p$ ?

# Kullback-Leibler divergence

How far off is our model  $q$  from the true distribution  $p$ ?

$$D_{\text{KL}} = \sum_{i=1}^n p_i \log \left( \frac{p_i}{q_i} \right)$$

## Activity

Calculate KL divergence for the following sample sizes:

- ▶ 5
- ▶ 10
- ▶ 20

$$D_{\text{KL}} = \sum_{i=1}^n p_i (\log(p_i) - \log(q_i))$$

- ▶ Average difference in log probability between  $p$  and  $q$

### Bonus

- ▶ What happens when you draw 1000 samples?
- ▶ What happens when our approximation  $q$  is exactly the same as  $p$ ?

# The problem with reality

We almost never know the true probability of events!

## What *do* we have

Typically we have data  $y_1, y_2, \dots, y_n$   
and some models (let's say two)

$$q, r$$

**So we can ask**

Which model seems closer to the true distribution  $p$ ?

$$D_{\text{KL}}(p, q) - D_{\text{KL}}(p, r) = -(E \log(q_i) - E \log(r_i))$$

## Comparing models $q$ and $r$

$$D_{\text{KL}}(p, q) - D_{\text{KL}}(p, r) = -(E \log(q_i) - E \log(r_i))$$

**Notice that we don't need  $p$  to compute this difference!**



# Deviance

$$D_{\text{KL}}(p, q) - D_{\text{KL}}(p, r) = -(E \log(q_i) - E \log(r_i))$$

We can plug in something proportional to the expected log likelihood:

$$E \log(q_i) \propto$$

$$D(q) = -2 \sum_{i=1}^n \log(q_i)$$

where  $D(q)$  is the **Deviance**

## Deviance in plain english

$$D(q) = -2 \sum_{i=1}^n \log(q_i)$$

A relative measure of divergence from the true distribution  $p$

- ▶ one deviance value is useless
- ▶ multiple values allow us to compare models

## How to calculate deviance

$$D(q) = -2 \sum_{i=1}^n \log(q_i)$$

Log likelihood:  $\sum_{i=1}^n \log(q_i)$

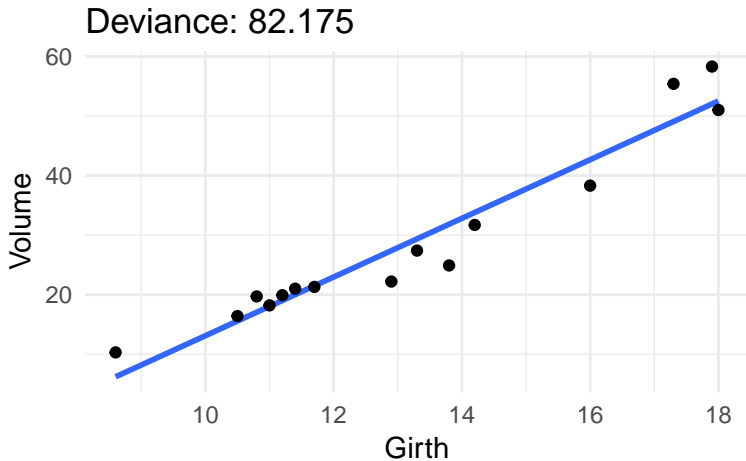
→ multiply log likelihood by -2.

\*demo

# The problem with Deviance

New predictors improve (reduce) deviance

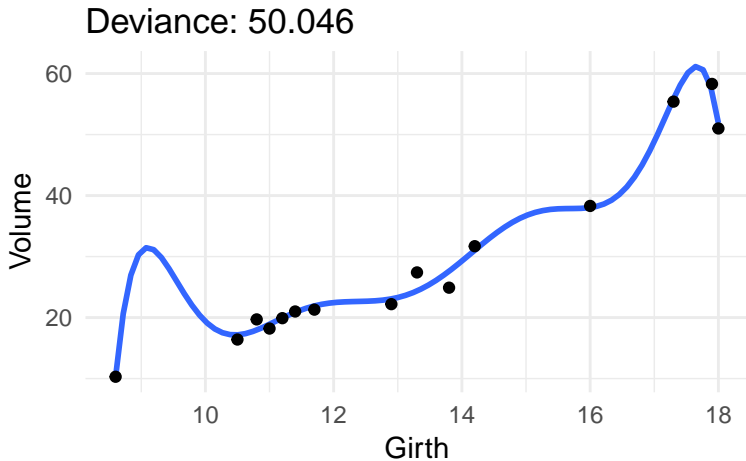
- ▶ same problem as  $R^2$



# The problem with Deviance

New predictors improve (reduce) deviance

- ▶ same problem as  $R^2$



## At the end of the day

We want to be close to the **truth** but not too close to our training **data**

# We want to make good predictions

In other words, we'd like low deviance for **new** observations

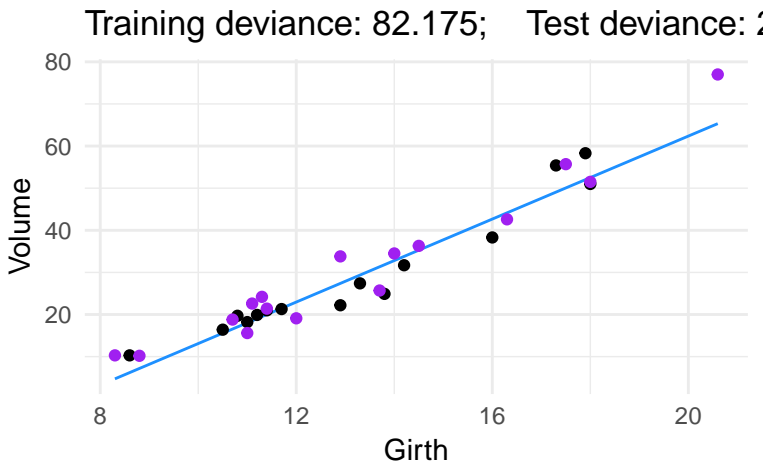
Deviance of training data  $(q_1, q_2, \dots)$ :

$$D_{\text{train}}(q) = -2 \sum_i \log(q_i)$$

Deviance of future data  $(\tilde{q}_1, \tilde{q}_2, \dots)$ :

$$D_{\text{test}}(q) = -2 \sum_i \log(\tilde{q}_i)$$

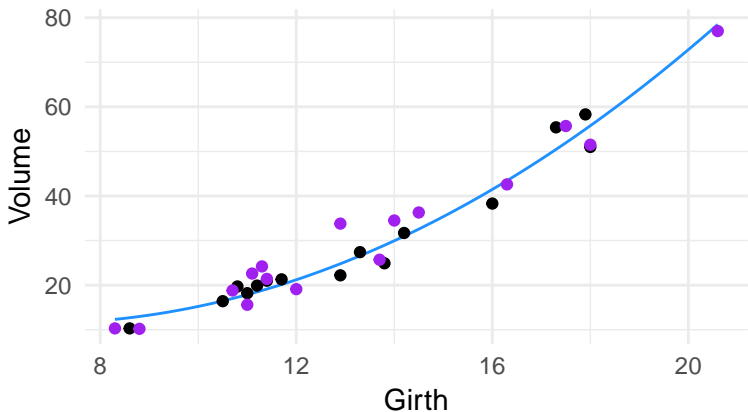
## Evaluating deviance of the training and test set





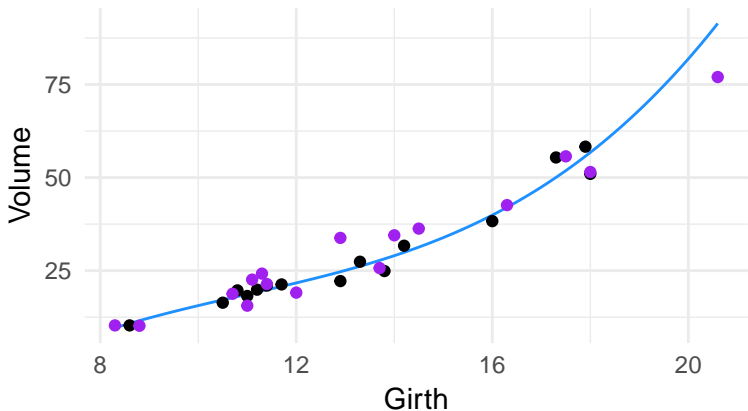
# Evaluating deviance of the training and test set

Training deviance: 72.624; Test deviance: 101.504



## Evaluating deviance of the training and test set

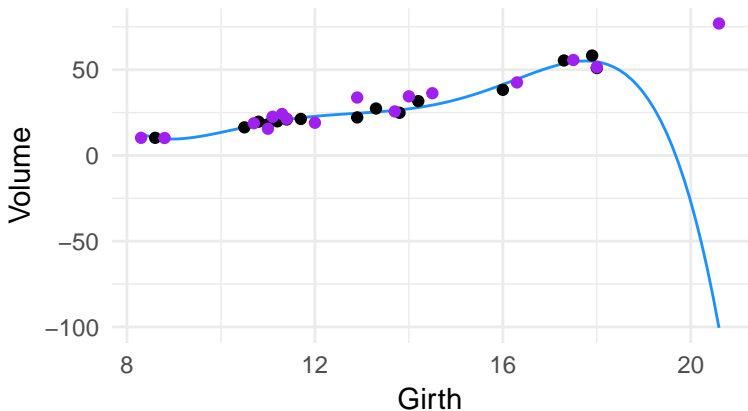
Training deviance: 70.792; Test deviance: 80.792



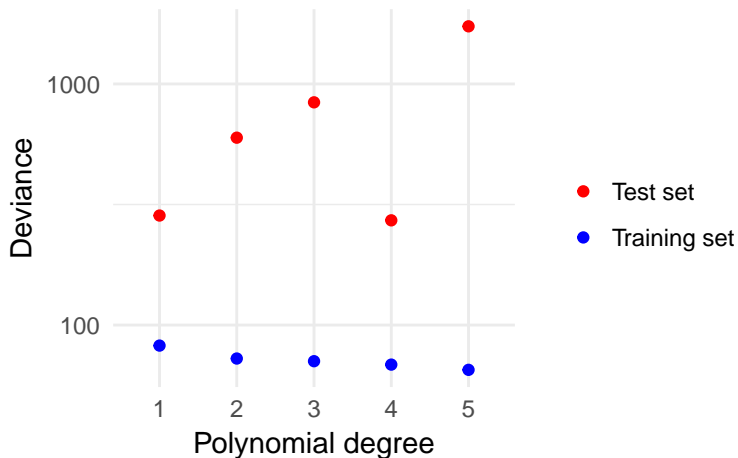


## Evaluating deviance of the training and test set

Training deviance: 65.188; Test deviance



# Training and test deviance across a range of model complexity



# Some problems with training vs. test splits

1. How to decide what goes where?
2. What if you have a small dataset?
3. What if your data are structured (e.g., by spatial location)

# Enter AIC

Instead of computing  $D_{\text{test}}$ , approximate with

## Akaike's information criterion

$$AIC = D_{\text{train}} + 2p$$

- ▶  $D_{\text{train}}$  is your training set deviance
- ▶  $p$  is the number of parameters in your model

“Better” models have lower  $AIC$

## Recap:

1. Over vs. underfitting
  2. Measured how close a model  $q$  is to truth  $p$  (KL divergence)
  3. Realized that since we don't know truth, we need a **relative** measure
  4. Learned that  $AIC$  is that relative measure
- how well can a model predict new data?

$$AIC = D_{\text{train}} + 2p$$