# Introduction to nonparametric regression: Least squares vs. Nearest neighbors

Patrick Breheny

October 30

# Introduction

- For the remainder of the course, we will focus on nonparametric regression and classification
- The regression problem involves modeling how the expected value of a response $y$ changes in response to changes in an explanatory variable $x$:

$$\mathbb{E}(y|x) = f(x)$$

- Linear regression, as its name implies, assumes a linear relationship; namely, that $f(x) = \beta_0 + \beta_1 x$

# Parametric vs. nonparametric approaches

- This reduction of a complicated function to a simple form with a small number of unknown parameters is very similar to the parametric approach to estimation and inference involving the unknown distribution function
- The nonparametric approach, in contrast, is to make as few assumptions about the regression function $f$ as possible
- Instead, we will try to use the data as much as possible to learn about the potential shape of $f$ – allowing $f$ to be very flexible, yet smooth

# Simple local models

- One way to achieve this flexibility is by fitting a different, simple model separately at every point $x_0$ in much the same way that we used kernels to estimate density
- As with kernel density estimates, this is done using only those observations close to $x_0$ to fit the simple model
- As we will see, it is possible to extend many of the same kernel ideas we have already discussed to smoothly “blend” these local models to construct a smooth estimate $\hat{f}$ of the relationship between $x$ and $y$

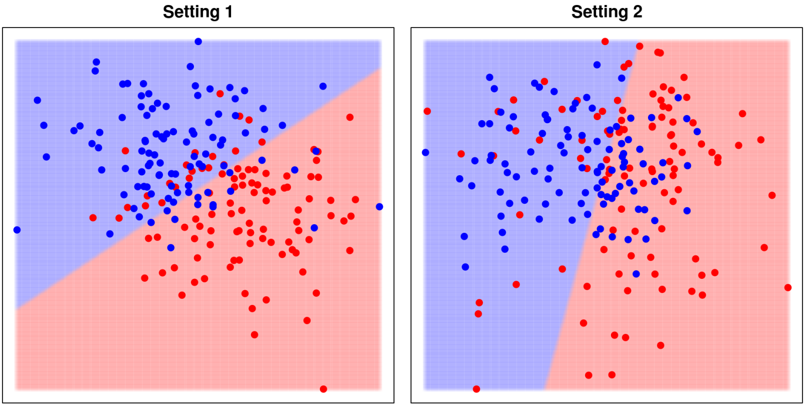
# Introduction

- Before we do so, however, let us get a general feel for the contrast between parametric and nonparametric classification by contrasting two simple, but very different, methods: the ordinary least squares regression model and the $k$-*nearest neighbor* prediction rule
- The linear model makes huge assumptions about the structure of the problem, but is quite stable
- Nearest neighbors is virtually assumption-free, but its results can be quite unstable
- Each method can be quite powerful in different settings and for different reasons

# Simulation settings

- To examine which method is better in which setting, we will simulate data from a simple model in which $y$ can take on one of two values: $-1$ or $1$
- The corresponding $x$ values are derived from one of two settings:
  - Setting 1: $x$ values are drawn from a bivariate normal distribution with different means for $y = 1$ and $y = -1$
  - Setting 2: A mixture in which 10 sets of means for each class $(1, -1)$ are drawn; $x$ values are then drawn by randomly selecting a mean from the appropriate class and then generating a random bivariate normal observation with that mean
- A fair competition between the two methods is then how well they do at predicting whether a future observation is $1$ or $-1$ given its $x$ values

# Linear model results



# Linear model remarks

- The linear model seems to classify points reasonably in setting 1
- In setting 2, on the other hand, there are some regions which seem questionable
- For example, in the lower left hand corner of the plot, does it really make sense to predict "blue" given that all of the nearby points are "red"?

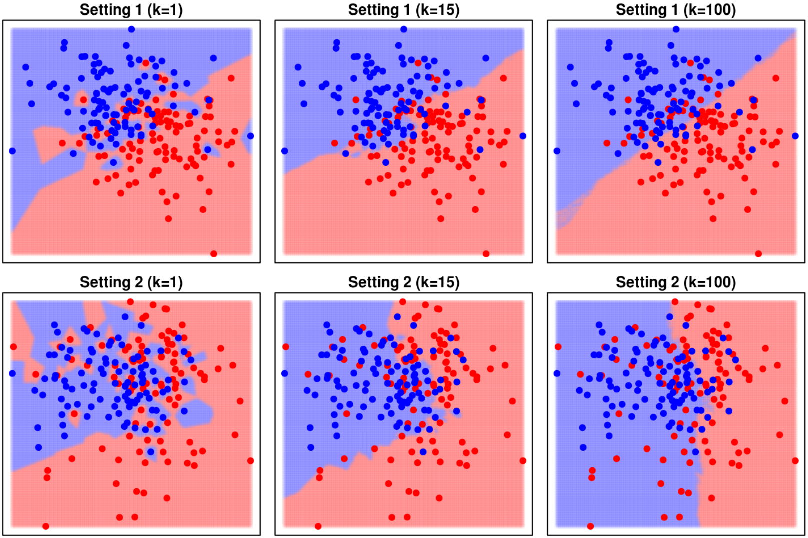
# Nearest neighbors

- Consider then a completely different approach in which we don't assume a model, a distribution, a likelihood, or anything about the problem: we just look at nearby points and base our prediction on the average of those points
- This approach is called the *nearest-neighbor* method, and is defined formally as

$$\hat{y}(\mathbf{x}) = \frac{1}{k} \sum_{\mathbf{x}_i \in N_k(\mathbf{x})} y_i,$$

where $N_k(\mathbf{x})$ is the neighborhood of $\mathbf{x}$ defined by its $k$ closest points in the sample

# Nearest neighbor results



# Nearest neighbor remarks

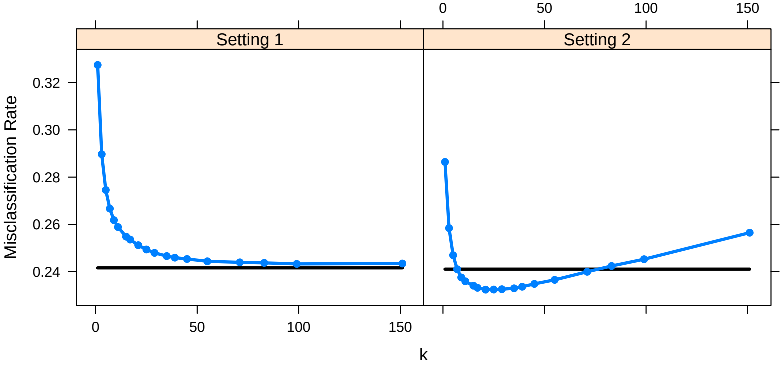
- Nearest neighbor seems not to perform terribly well in setting 1, as its classification boundaries are unnecessarily complex and unstable
- On the other hand, the method seemed perhaps better than the linear model in setting 2, where a complex and curved boundary seems to fit the data better
- Furthermore, the choice of $k$ plays a big role in the fit, and the optimal $k$ might not be the same in settings 1 and 2

# Inference

- Of course, it is potentially misleading to judge whether a method is better simply because it fits the sample better
- What matters, of course, is how well its predictions generalize to new samples
- Thus, consider generating 100 data sets of size 200, fitting each model, and then measuring how well each method does at predicting 10,000 new, independent observations

# Simulation results

Black line = least squares; blue line = nearest neighbors



# Remarks

- In setting 1, linear regression was always better than nearest neighbors
- In setting 2, nearest neighbors was usually better than linear regression
- However, it wasn't *always* better than linear regression – when $k$ was too big or too small, the nearest neighbors method performed poorly
- In setting 1, the bigger $k$ was, the better; in setting 2, there was a "Goldilocks" value of $k$ (about 25) that proved optimal in balancing the bias-variance tradeoff

# Conclusions

Thus,

- Fitting an ordinary linear model is rarely the best we can do
- On the other hand, nearest-neighbors is rarely stable enough to be ideal, even in modest dimensions, unless our sample size is very large (recall the curse of dimensionality)

# Conclusions (cont'd)

- These two methods stand on opposite sides of the methodology spectrum with regard to assumptions and structure
- The methods we will discuss for the remainder of the course involve bridging the gap between these two methods – making linear regression more flexible, adding structure and stability to nearest neighbor ideas, or combining concepts from both
- As with kernel density estimation, the main theme that emerges is the need to apply methods that bring the right mix of flexibility and stability that is appropriate for the data