Lecture 13. Nonparametric GLMs

Nan Ye

School of Mathematics and Physics
University of Queensland

Nonparametric Models

Parametric models

- Fixed structure and number of parameters.
- Represent a fixed class of functions.

Nonparametric models

- Flexible structure where the number of parameters usually grow as more data becomes available.
- The class of functions represented depends on the data.
- Not models without parameters, but nonparametric in the sense that they do not have fixed structures and numbers of parameters as in parametric models.

This Lecture

- k-NN
- LOESS
- Splines

k-NN Regression

Algorithm

- Training set is $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$.
- To compute $\mathbb{E}(Y \mid \mathbf{x})$ for any $\mathbf{x}$
  - $N_k(\mathbf{x}) \leftarrow$ nearest k training examples.
  - Predict the average response for the examples in $N_\alpha(\mathbf{x})$.

Effect of k

- Training error is zero when $k = 1$, and approximately increases as k increases.
- However, the fitted 1-NN model is often not smooth and does not work well on test data.
- Cross-validation can be used to choose a suitable k.

Remarks

- k-NN is data inefficient
  - For high-dimensional problems, the amount of data required for good performance is often huge.
- k-NN is computationally inefficient
  - Naively, predicting on m test examples requires $O(nmk)$ time.
  - This can be improved, but still k-NN is very slow.

LOESS (LOcal regrESSion)

Idea

- Training set is $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$.
- To compute $\mathbb{E}(Y \mid \mathbf{x})$ for any $\mathbf{x}$
  - $N_\alpha(\mathbf{x}) \leftarrow$ nearest $n\alpha$ training examples.
  - Perform a weighted linear regression using $N_\alpha(\mathbf{x})$.
  - Evaluate the fitted linear model at $\mathbf{x}$.
- The locality parameter α controls the neighborhood size.

Details

- Local weighted linear regression is as follows

$$\theta = \arg\min_{\beta} \sum_{(\mathbf{x}', y') \in N_\alpha(\mathbf{x})} w(\|\mathbf{x} - \mathbf{x}'\|)(y' - \beta^\top \mathbf{x}')^2,$$

- The weight function w is defined by

$$w(d) = \left(1 - \frac{d^3}{M^3}\right)^3,$$

where $M = \max(1, \alpha)^{1/p} \max_{(\mathbf{x}', y') \in N_\alpha(\mathbf{x})} \|\mathbf{x} - \mathbf{x}'\|$ is the scaled maximum distance.

Effect of α

- If α is very small, the neighborhood may have too few points, for the weighted least squares problem to have a unique solution.
- In general, a smaller α makes the fitted surface more wiggly.
- As $\alpha \to \infty$, we have $w(d) \to 1$, and θ becomes the OLS parameter. Thus LOESS converges to OLS as $\alpha \to \infty$.

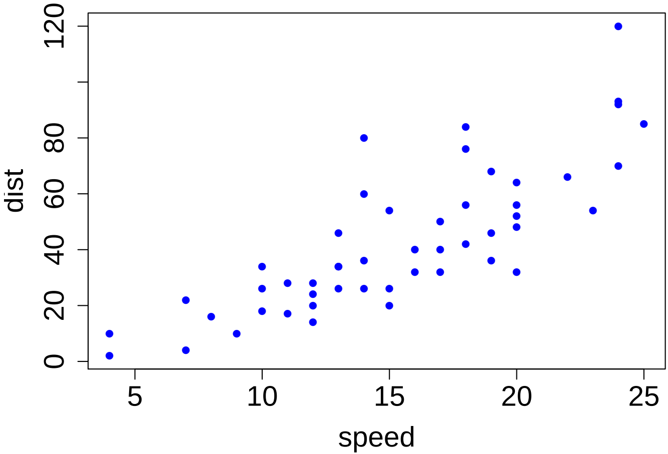
LOESS with higher degree terms

- We can add higher degree terms like quadratic terms $x_i x_j$ before we perform regression.
- This can be helpful if the linear predictor does not work well.

Data

```
> head(cars)
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
> dim(cars)
[1] 50  2
```

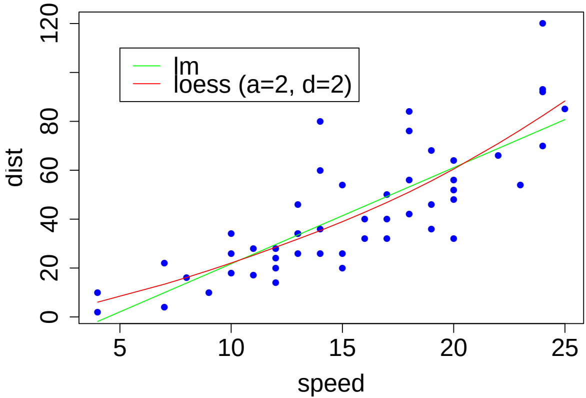
Scatterplot



LOESS in R

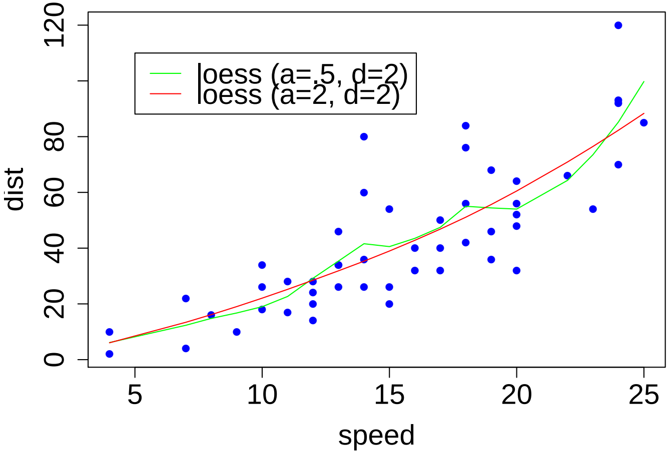
```
a = 2
deg = 2
fit.loess <- loess(dist ~ speed, cars, span=a, degree=deg)
```

Comparison of OLS and LOESS



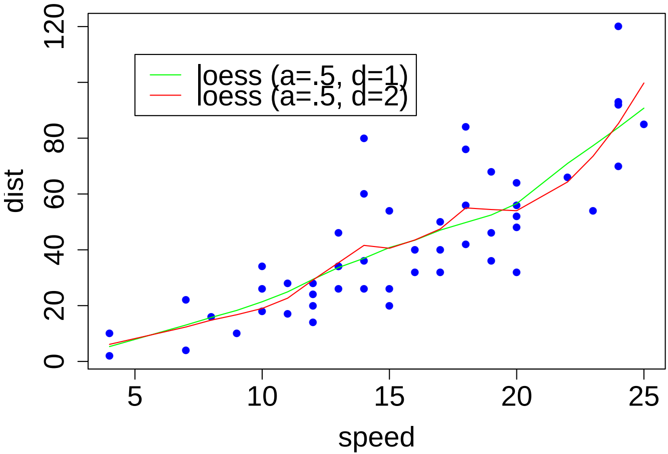
- The linearity assumption of OLS is rigid and does not adapt to the data's complexity.
- LOESS is capable of adapting to the data's complexity through local regression, and better fits the data than OLS.

Effect of α



Smaller α leads to a more wiggly fit.

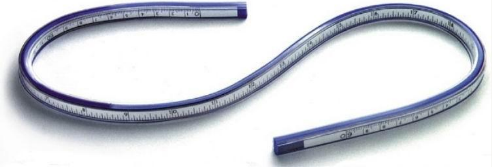
Effect of degree



Higher degree leads to a more wiggly fit.

Splines

- A flat spline is a device used for drawing smooth curves.



- A spline is a *smooth* piecewise polynomial function.

Spline, order, and knots

- A function $f : \mathbf{R} \to \mathbf{R}$ is a spline of order k with knots at $t_1 < \ldots < t_m$ if
  - $f(x)$ is a polynomial of degree k on each of the interval $(-\infty, t_1], [t_1, t_2], \ldots, [t_m, \infty)$, and
  - its i-th derivative $f^{(i)}(x)$ is continuous at each knot for each $i = 0, \ldots, k-1$.
- The cubic splines ($k = 3$) are most commonly used.
- *Natural splines* are linear beyond t_1 and t_m.

Truncated power basis

- An order-k spline with knots $t_1, \ldots, t_m$ is a linear combination of the following $k + m + 1$ basis functions

$$h_1(x) = 1,\ h_2(x) = x, \ldots,\ h_{k+1}(x) = x^k,$$
$$h_{k+1+j}(x) = (x - t_j)_+^k, \quad j = 1, \ldots, m,$$

  where $(x)_+ = \max(0, x)$ is the positive part function.
- These basis functions are called the truncated power basis.

Spline regression as linear regression

- Training data: $(x_1, y_1), \ldots, (x_n, y_n) \in \mathbf{R} \times \mathbf{R}$.
- Given knots $t_1, \ldots, t_m$, an order k spline is fitted by minimizing

$$\hat{\beta} = \sum_{i=1}^{n} (\beta^\top \mathbf{z}_i - y_i)^2,$$

  where $\mathbf{z}_i = (h_1(x_i), \ldots, h_{k+1+m}(x_i))$.
- The fitted spline is

$$f(x) = \sum_i \hat{\beta}_i h_i(x).$$

- The knots can be chosen in a data-dependent way (e.g. equally spaced between min and max x).

What You Need to Know

- Nonparametric models can adapt to data's complexity.
- k-NN: averaging over a neighborhood.
- LOESS: weighted linear regression over a neighborhood.
- Splines: fit smooth piecewise polynomials.