A BRIEF INTRODUCTION TO GAUSSIAN PROCESSES

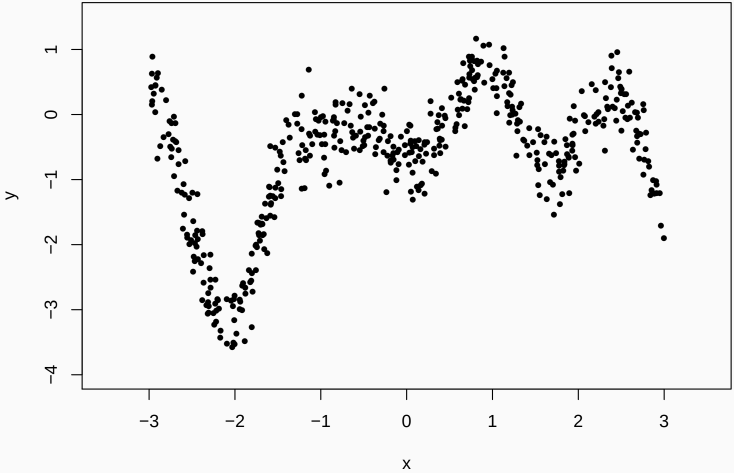
Colin Rundel

Duke - “What if?” Focus Cluster - 2015

Duke University
Department of Statistical Science

APPROACHES TO REGRESSION

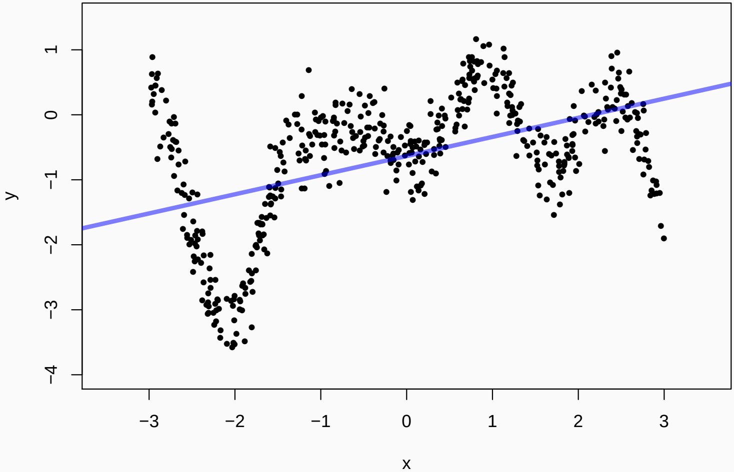
EXAMPLE DATA



SIMPLE LINEAR REGRESSION

Model: $y = \beta_0 + \beta_1 x$

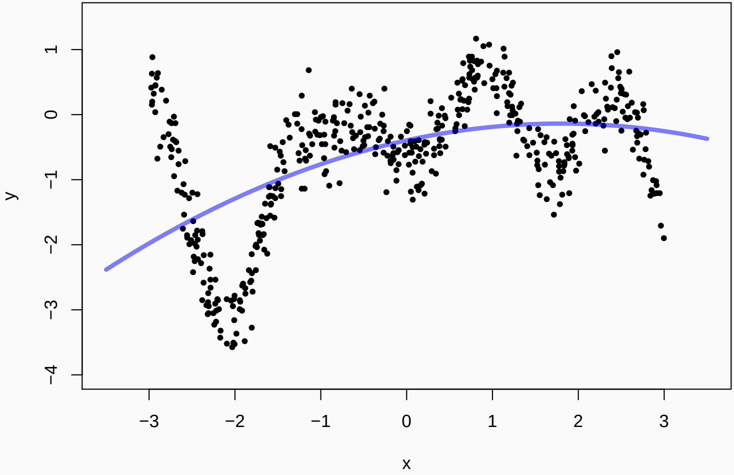
R: `l = lm(y~x)`



POLYNOMIAL REGRESSION

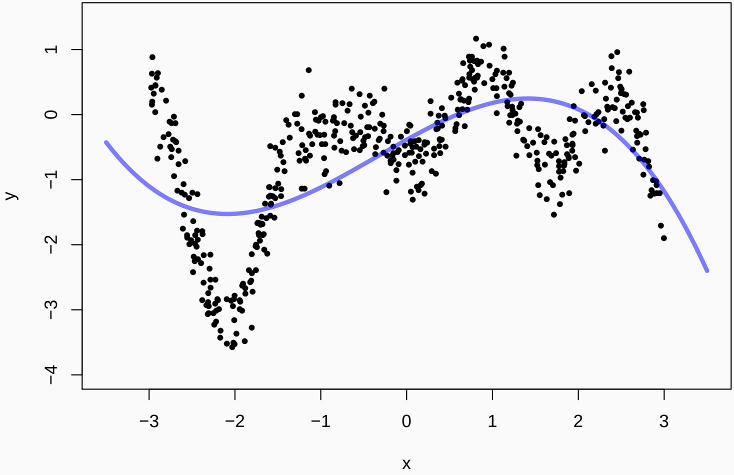
Model: $$y = \beta_0 + \beta_1 x + \beta_2 x^2$$

R:
```
l = lm(y~poly(x,2))
```



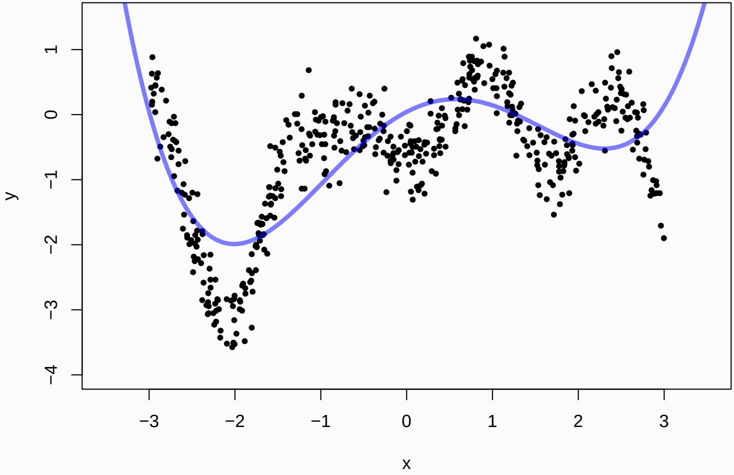
Model: $$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$$

R: `l = lm(y~poly(x,3))`



Model: $$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4$$

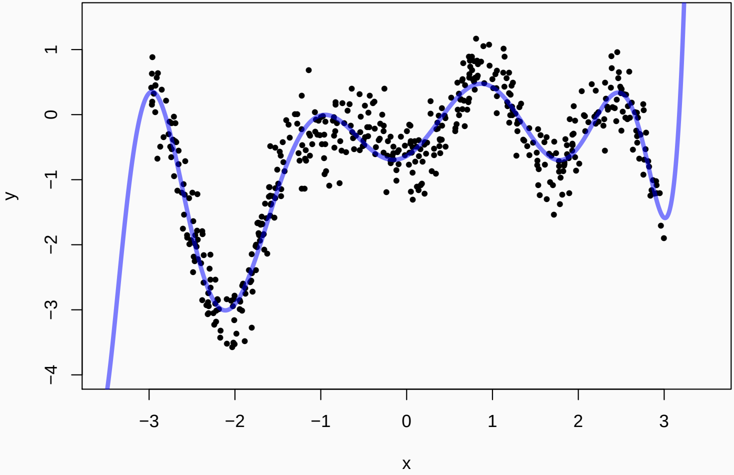
R: `l = lm(y~poly(x,4))`



POLYNOMIAL REGRESSION

Model: $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_{10} x^{10}$

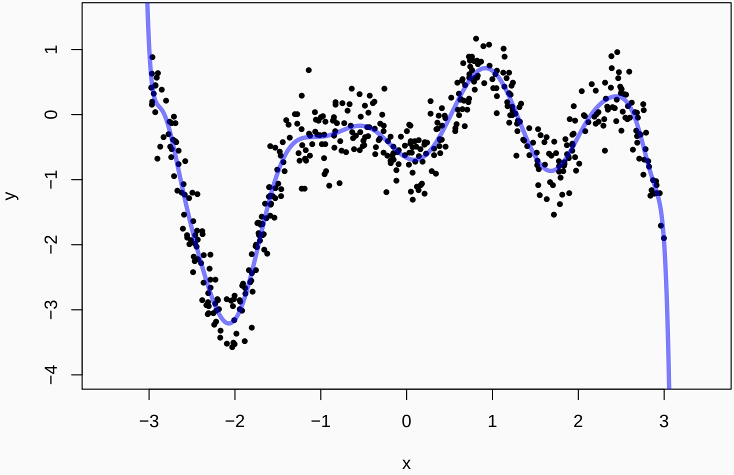
R: `l = lm(y~poly(x,10))`



POLYNOMIAL REGRESSION

Model: $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_{20} x^{20}$

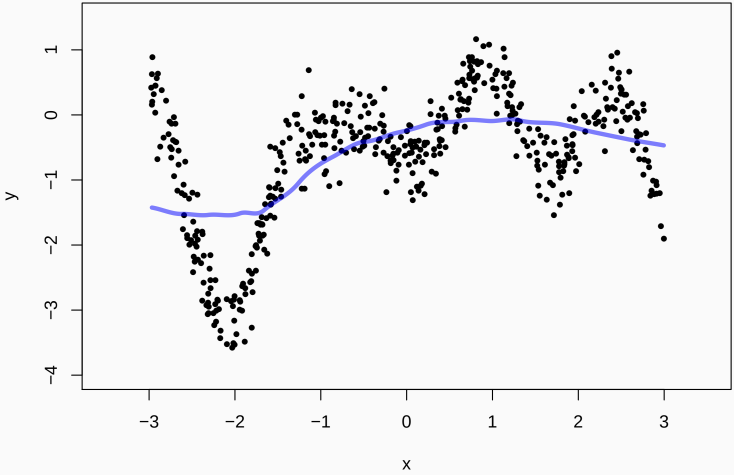
R: `l = lm(y~poly(x,20))`



LOCAL REGRESSION (LOESS / LOWESS)

Model: (non-parametric)

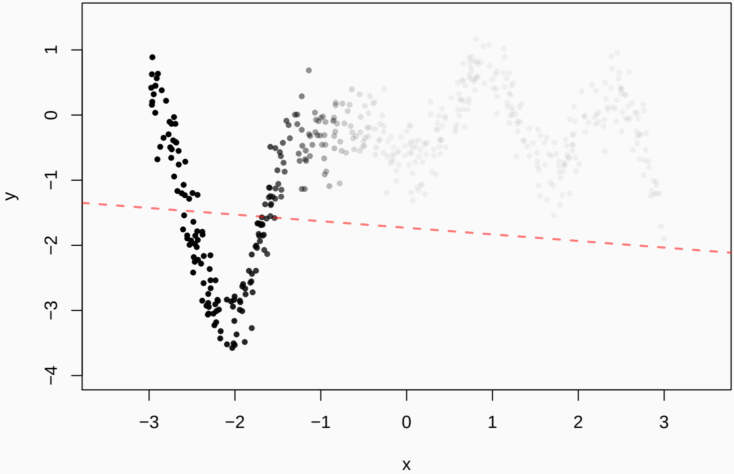
R: `l = loess(y~x, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

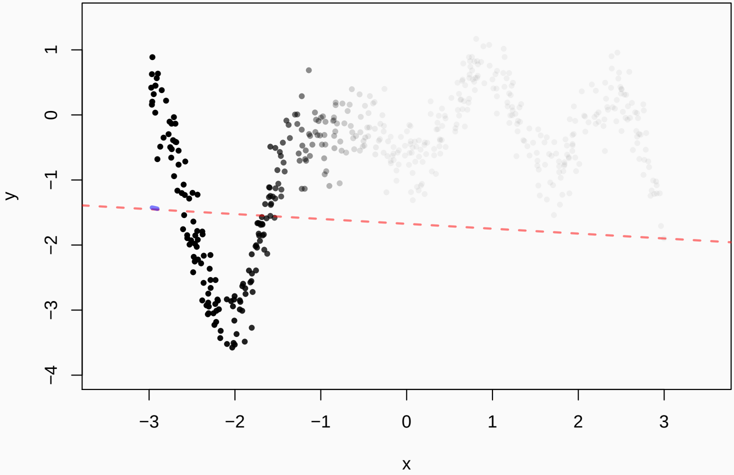
R:
```
l = loess(y~x, span=0.5, degree=1)
```



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

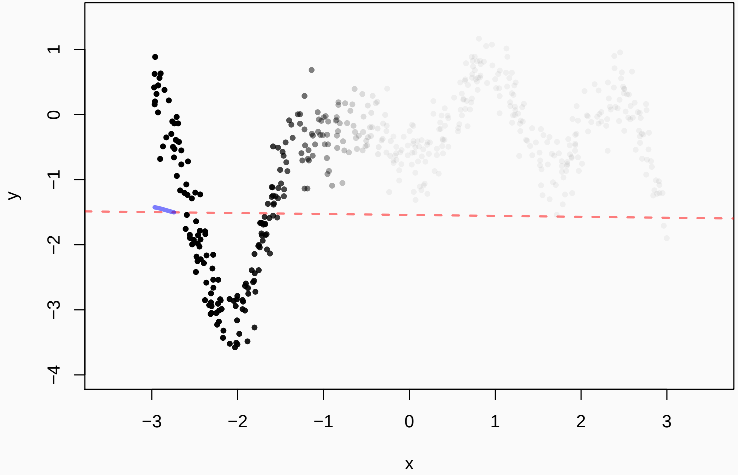
R: `l = loess(y~x, span=0.5, degree=1)`



Model: (non-parametric)

R:

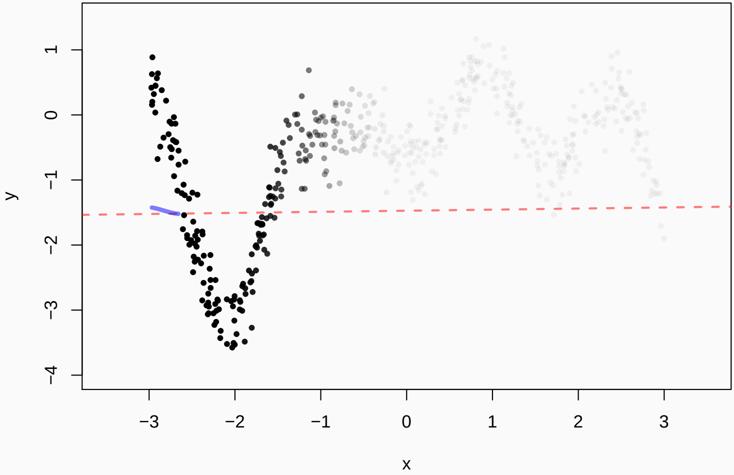
```
l = loess(y~x, span=0.5, degree=1)
```



LOCAL REGRESSION - HOW DOES IT WORK?

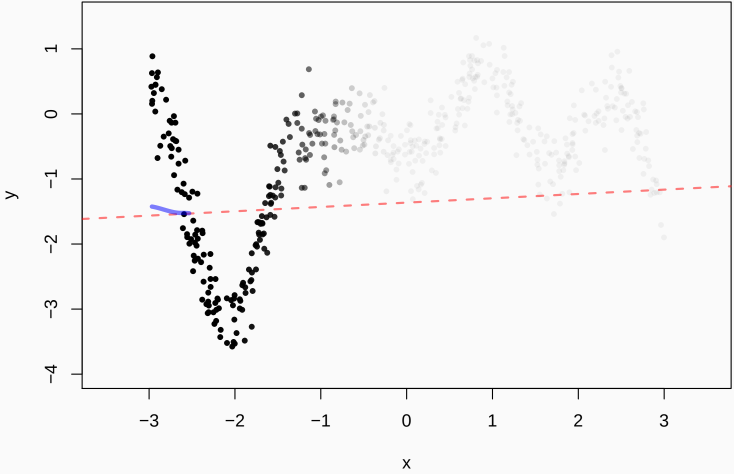
Model: (non-parametric)

R: `l = loess(y~x, span=0.5, degree=1)`



Model: (non-parametric)

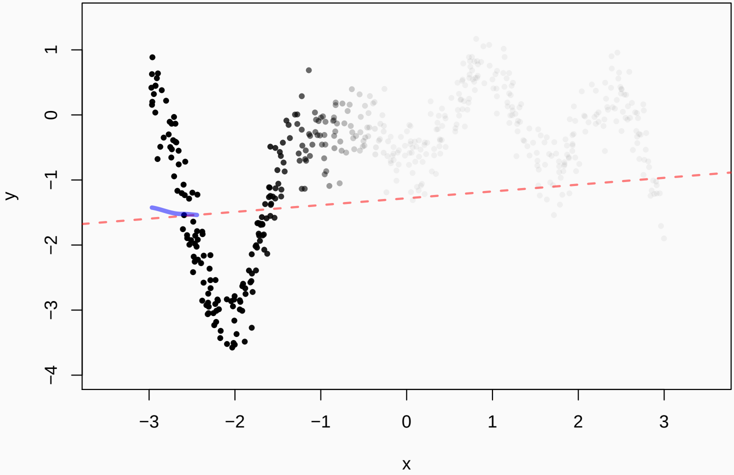
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

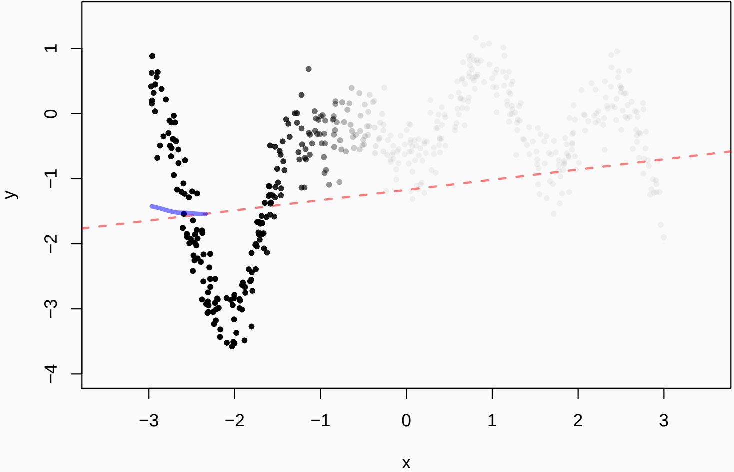
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

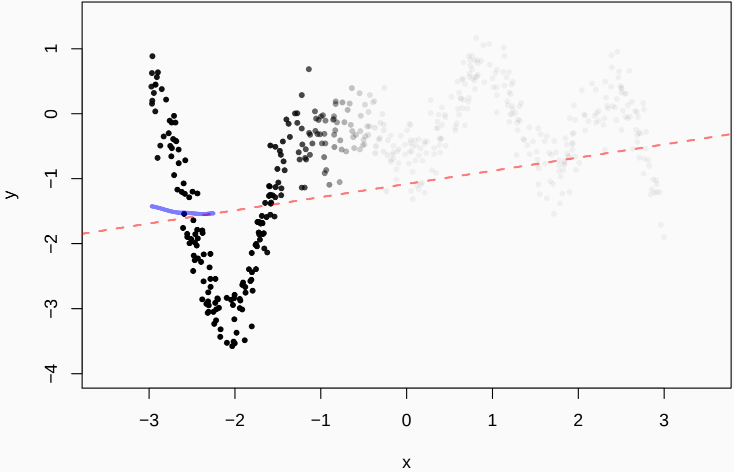
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

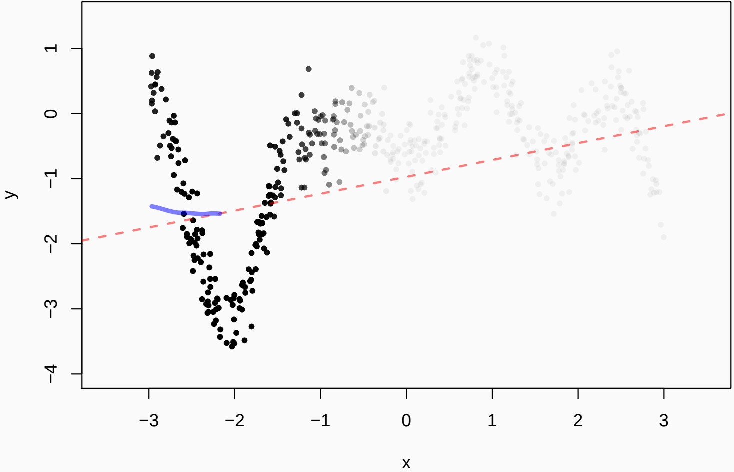
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

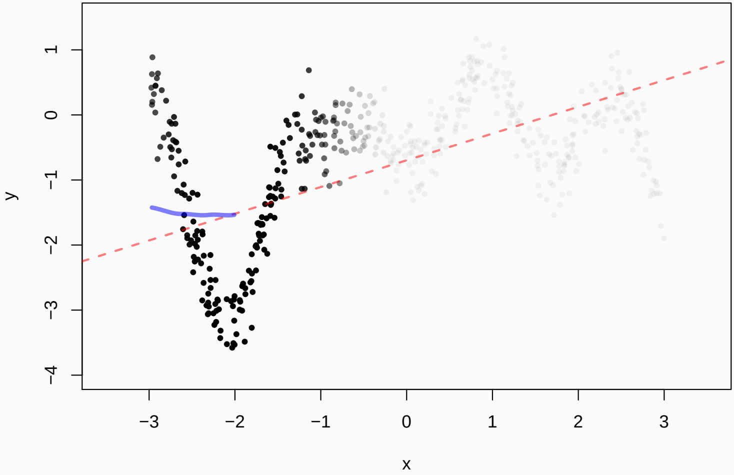
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

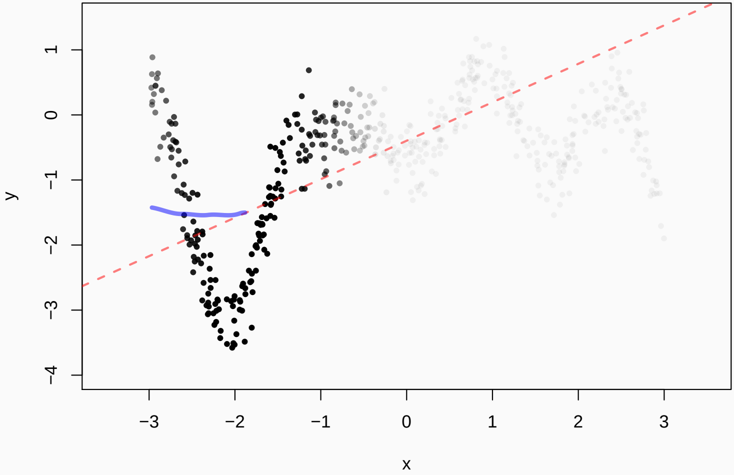
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

R: `l = loess(y~x, span=0.5, degree=1)`

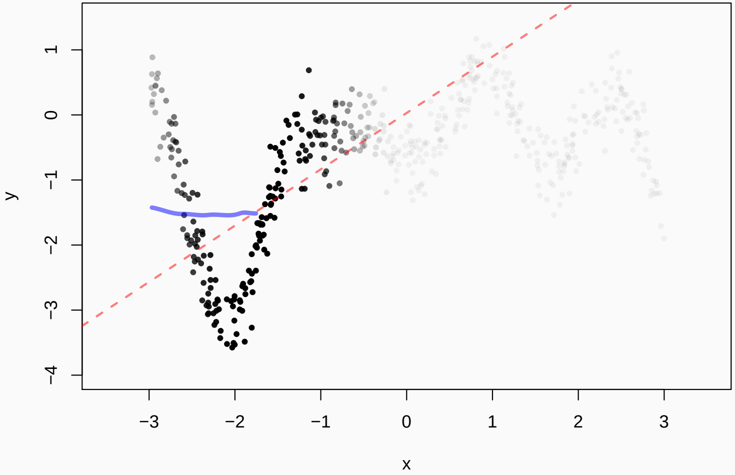


LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

R:

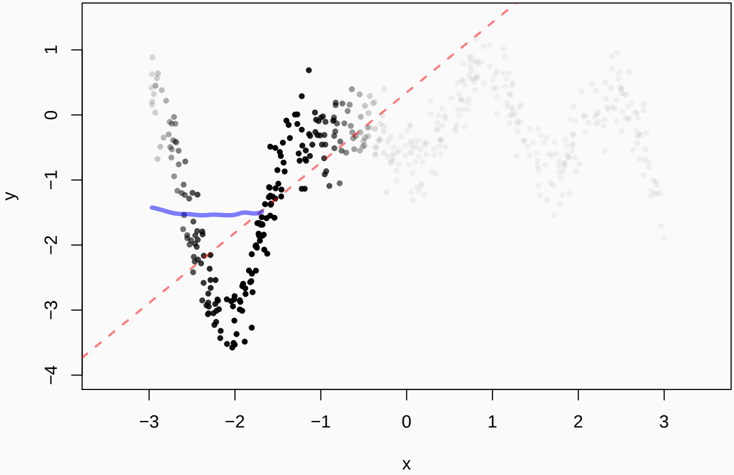
```
l = loess(y~x, span=0.5, degree=1)
```



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

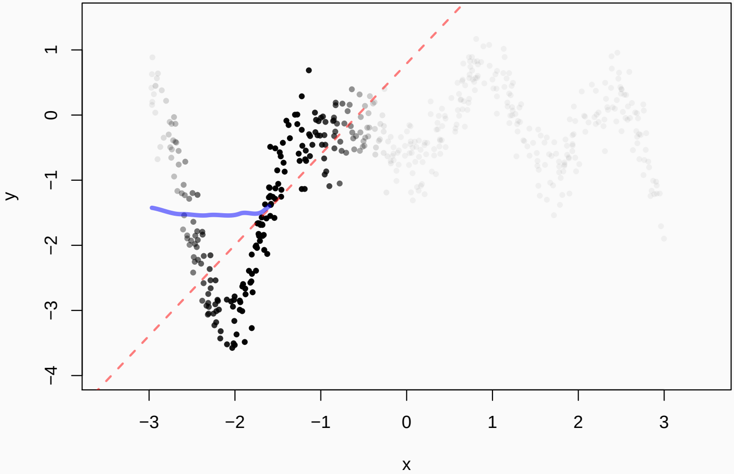
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

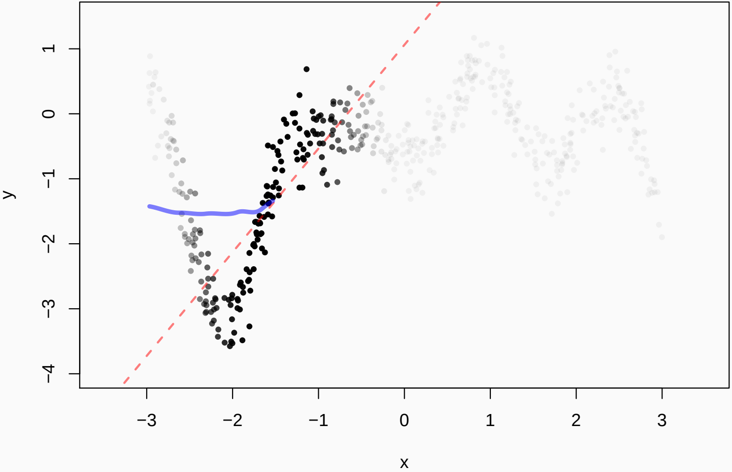
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

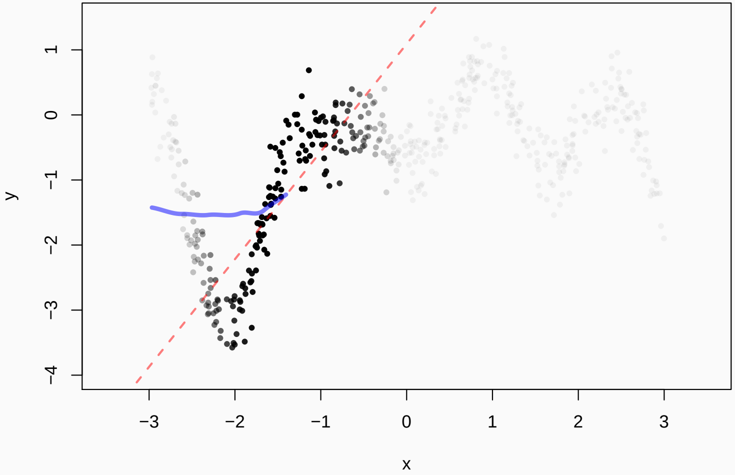
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

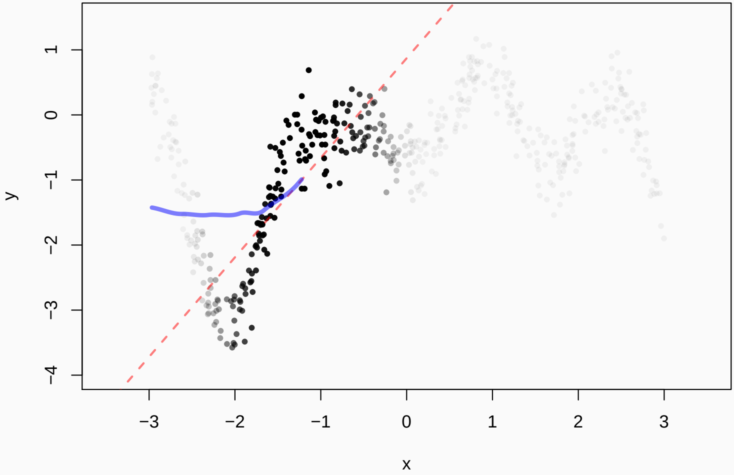
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

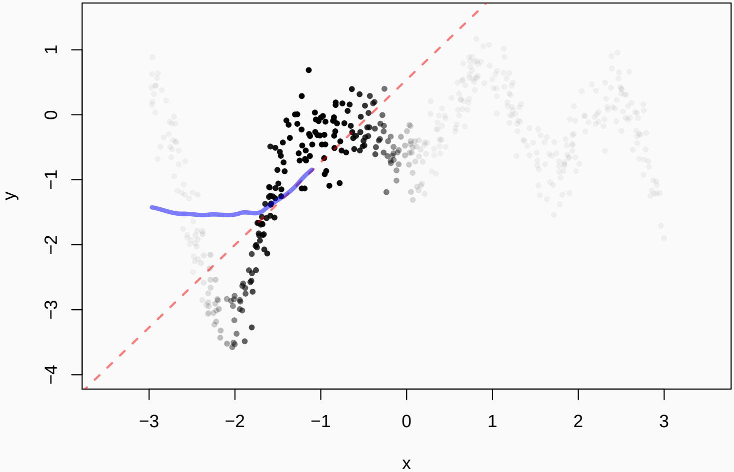
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

R: `l = loess(y~x, span=0.5, degree=1)`

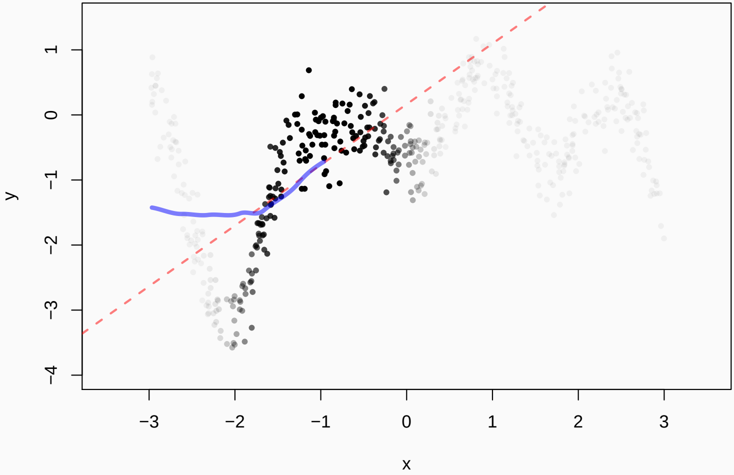


LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

R:

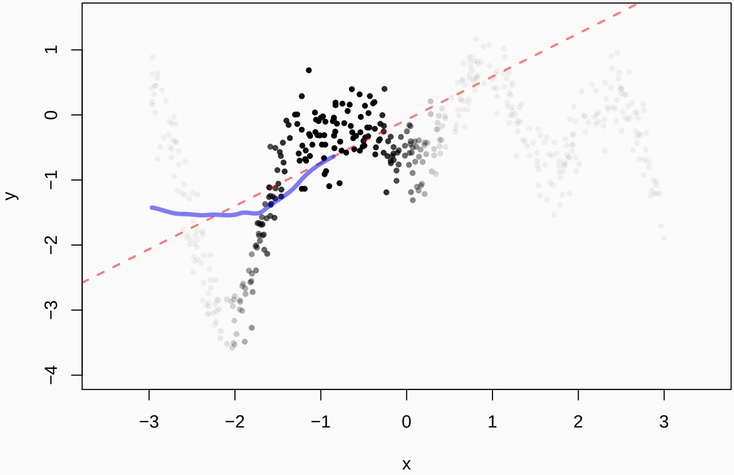
```
l = loess(y~x, span=0.5, degree=1)
```



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

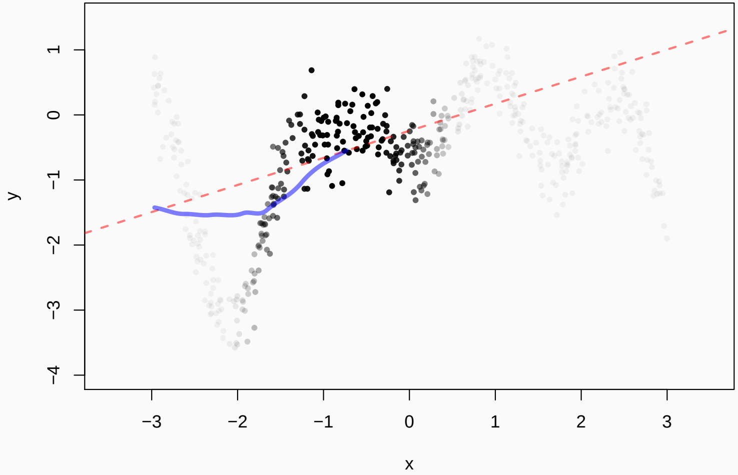
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

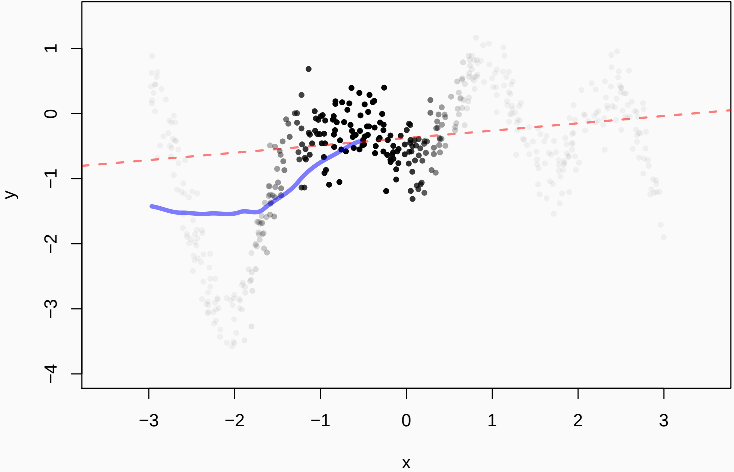
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

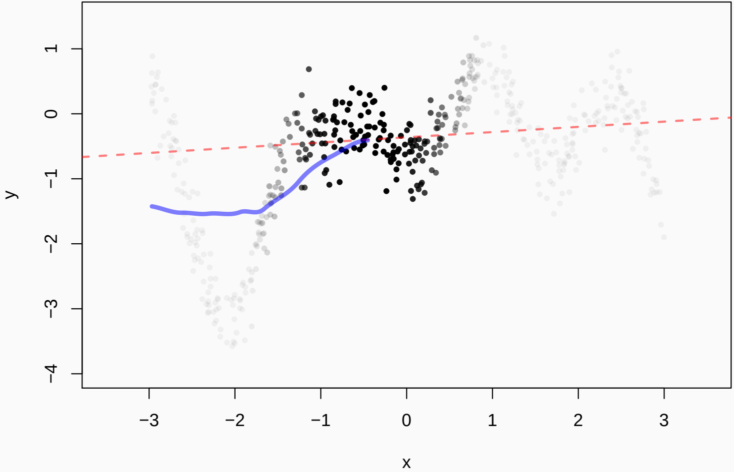
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

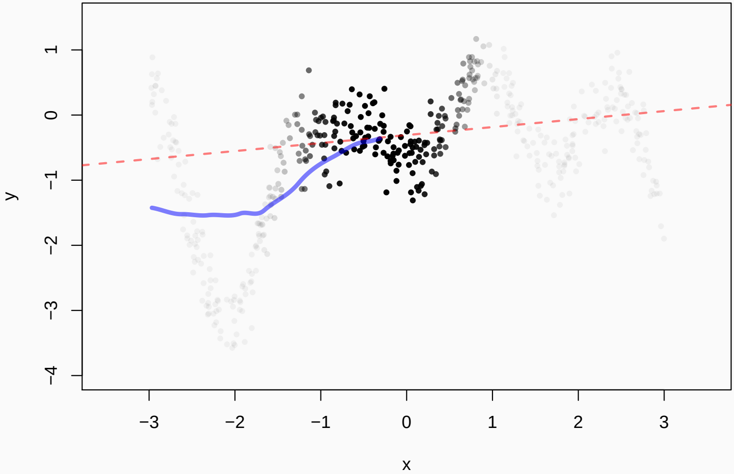
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

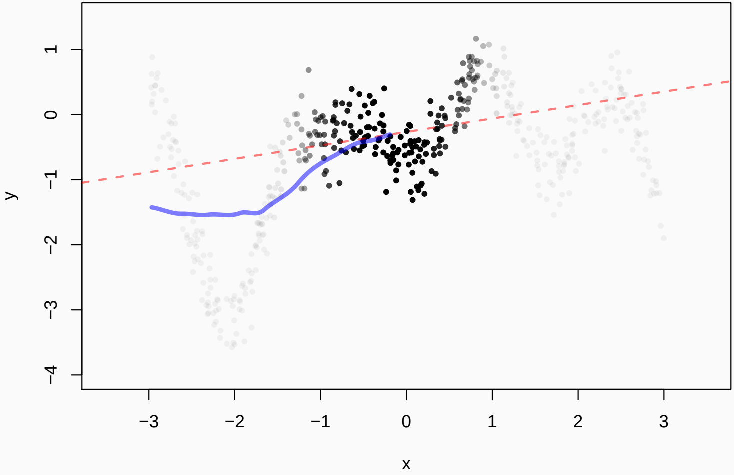
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

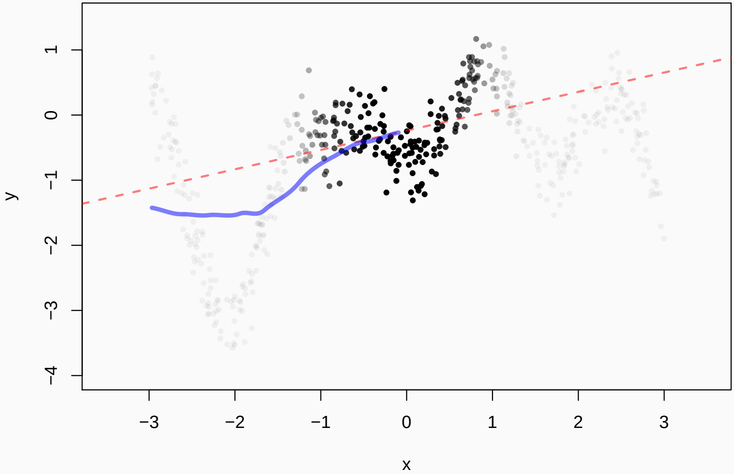
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

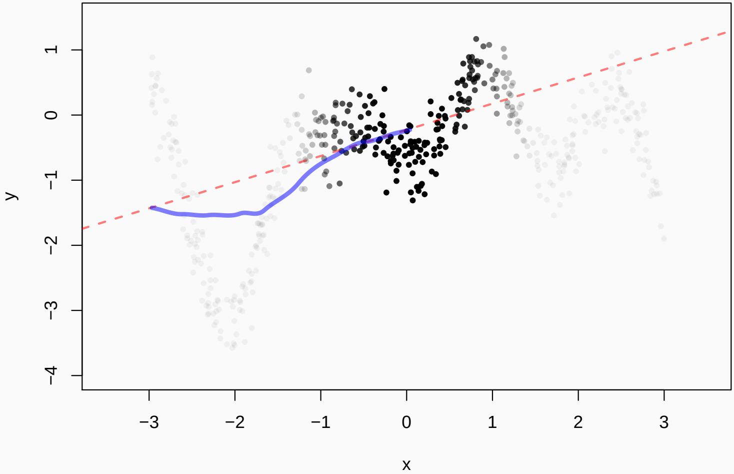
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

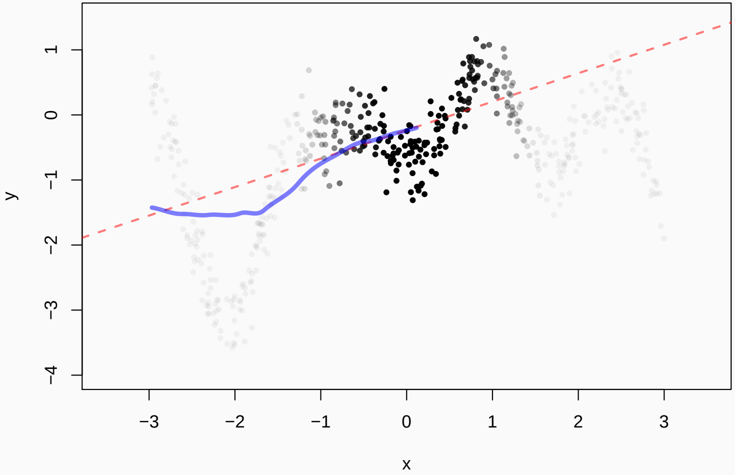
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

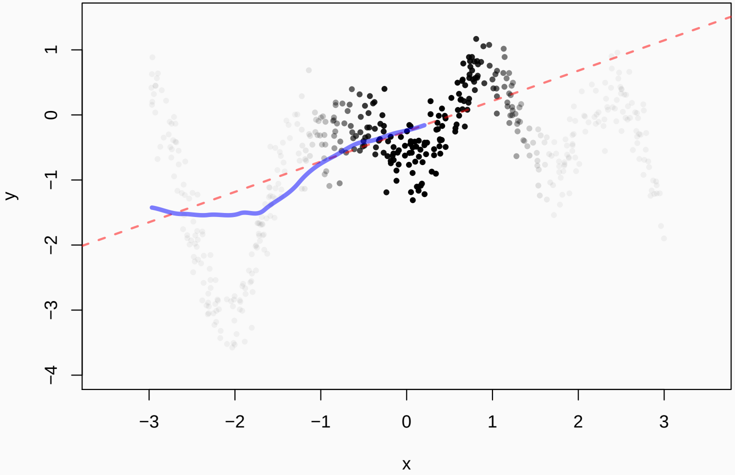
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

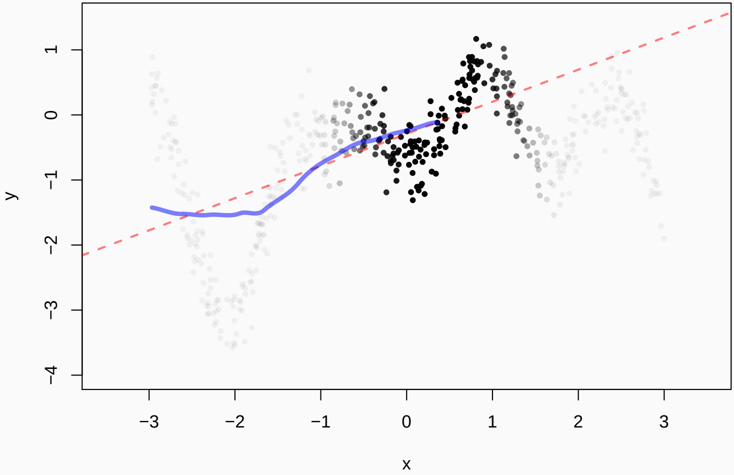
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

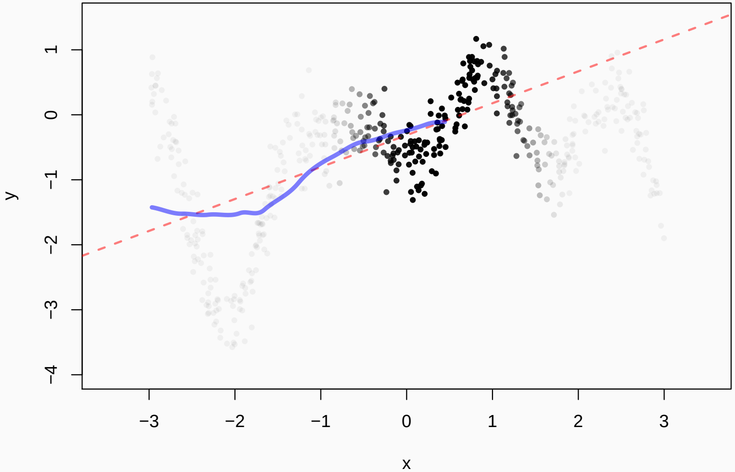
Model: (non-parametric)

R: `l = loess(y~x, span=0.5, degree=1)`



Model: (non-parametric)

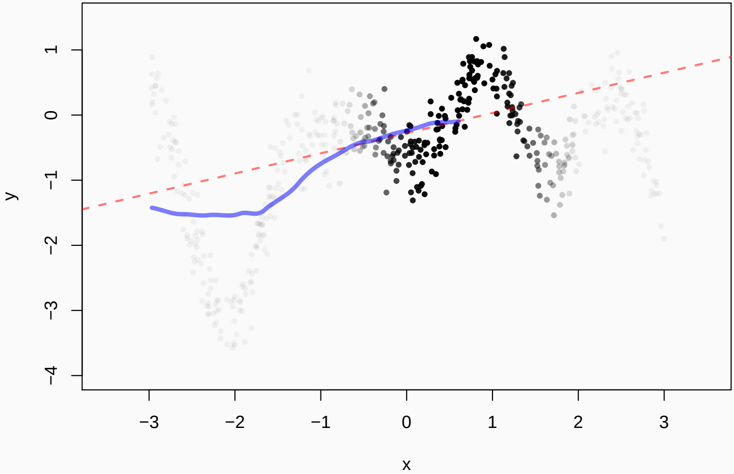
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

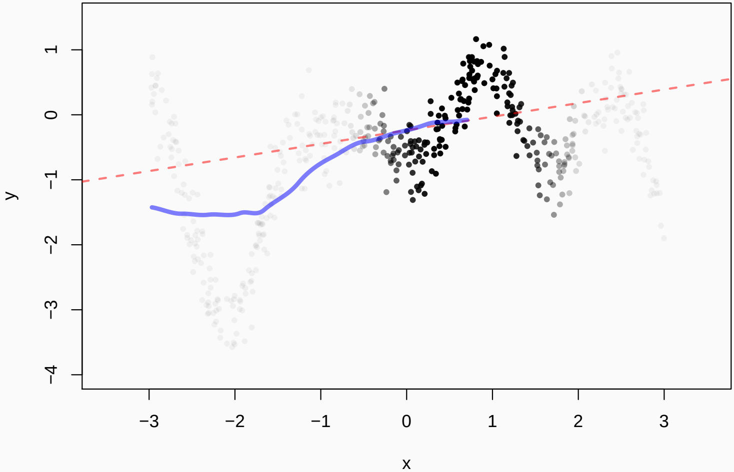
Model: (non-parametric)

R: `l = loess(y~x, span=0.5, degree=1)`



Model: (non-parametric)

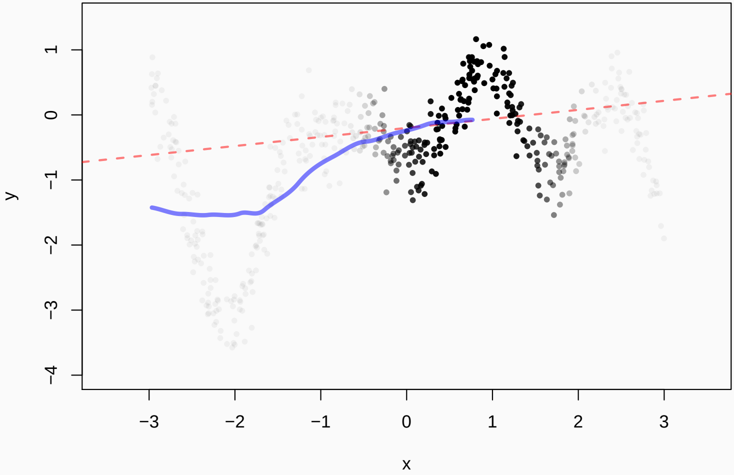
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

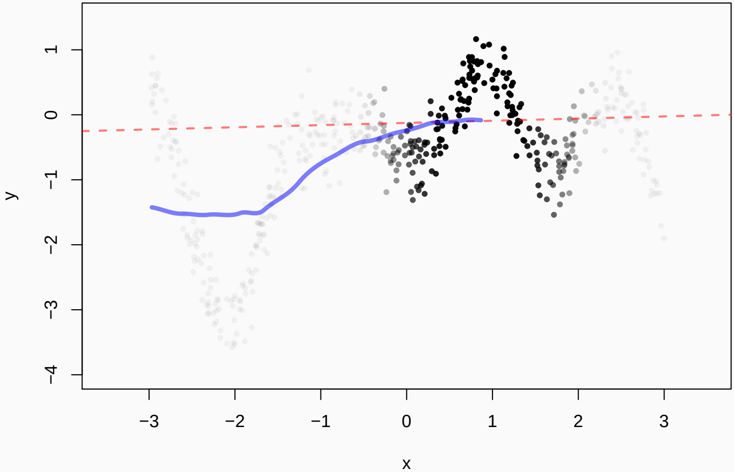
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

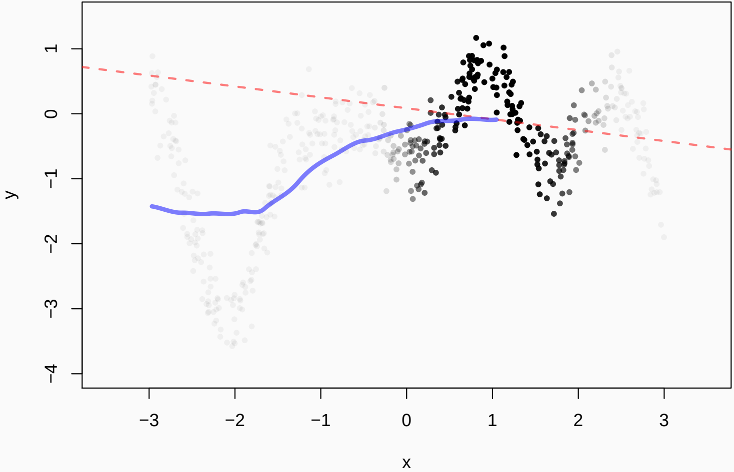
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

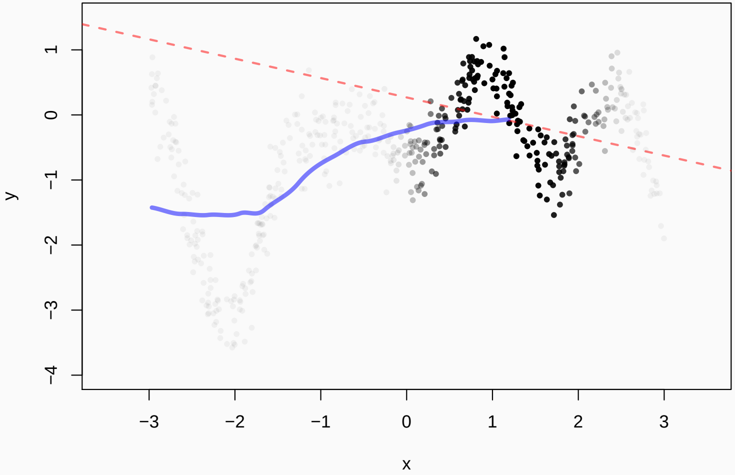
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

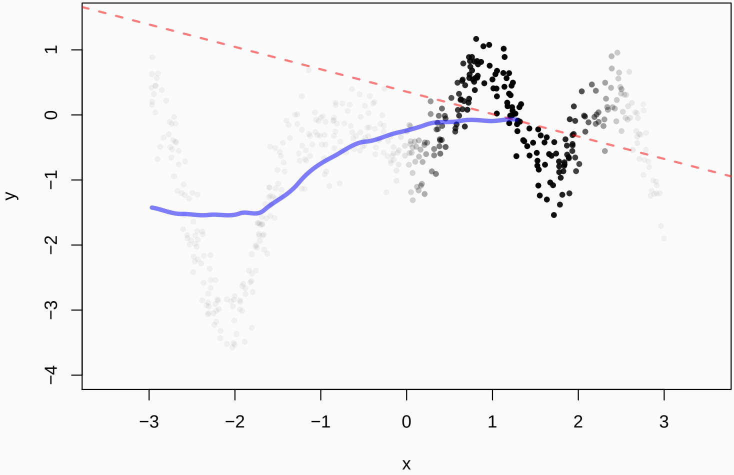
R:
```
l = loess(y~x, span=0.5, degree=1)
```



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

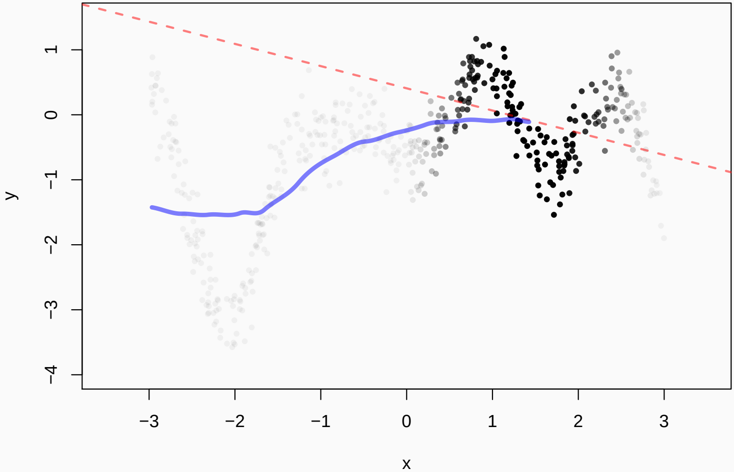
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

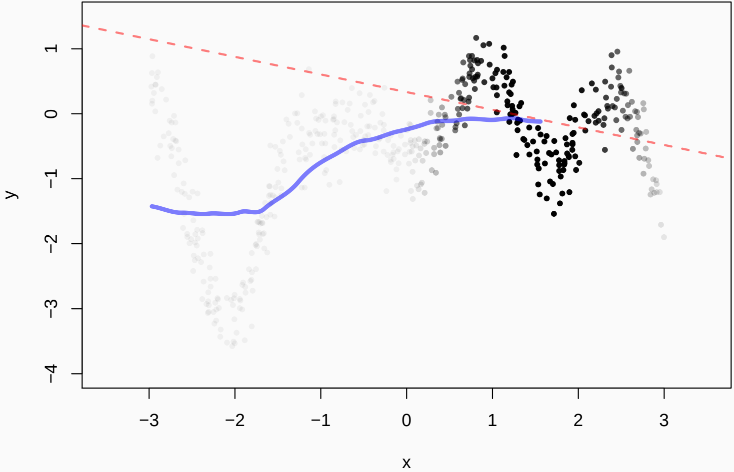
Model: (non-parametric)

R: `l = loess(y~x, span=0.5, degree=1)`



Model: (non-parametric)

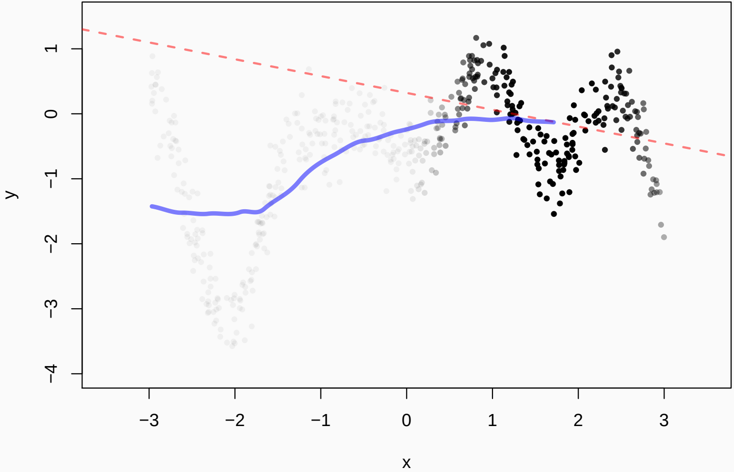
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

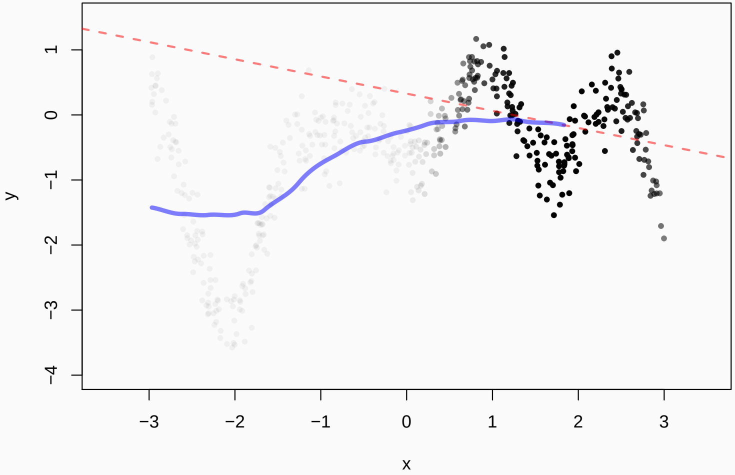
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

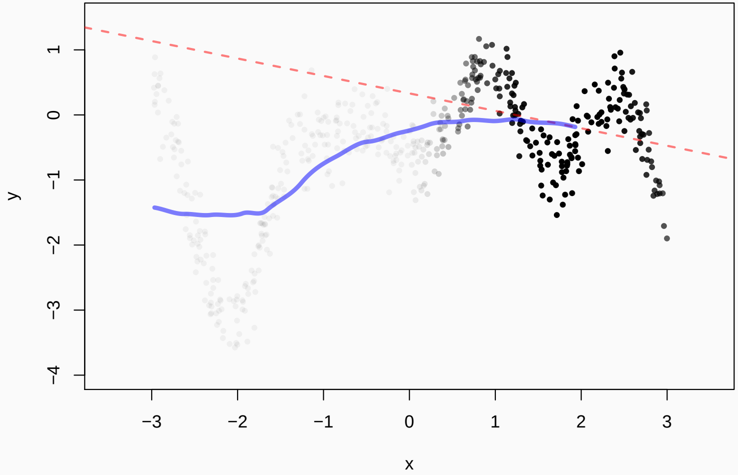
R:
```
l = loess(y~x, span=0.5, degree=1)
```



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

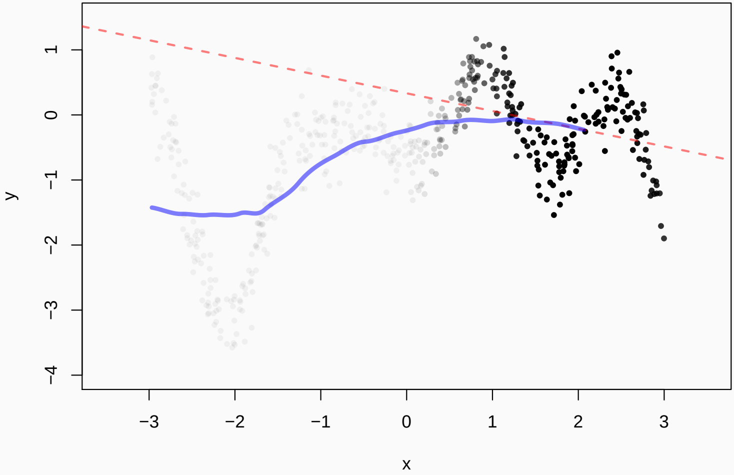
R:
```
l = loess(y~x, span=0.5, degree=1)
```



LOCAL REGRESSION - HOW DOES IT WORK?

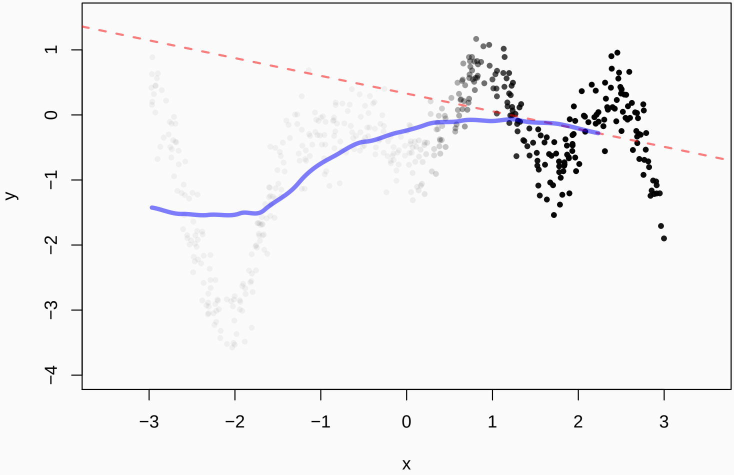
Model: (non-parametric)

R: `l = loess(y~x, span=0.5, degree=1)`



Model: (non-parametric)

R: `l = loess(y~x, span=0.5, degree=1)`

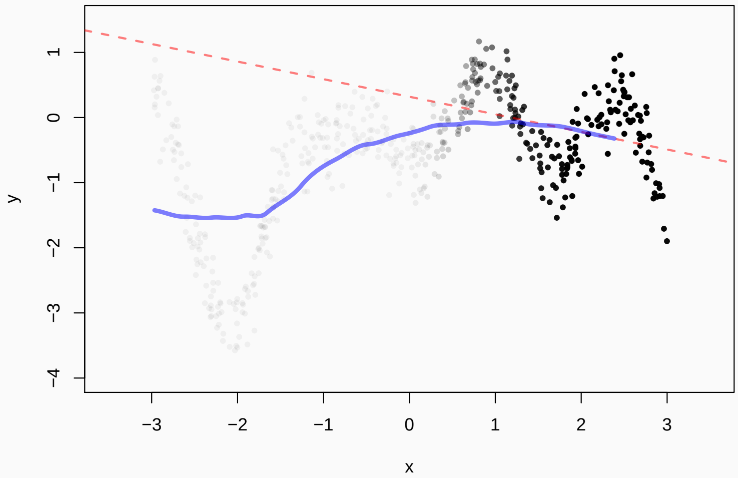


LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

R:

```
l = loess(y~x, span=0.5, degree=1)
```

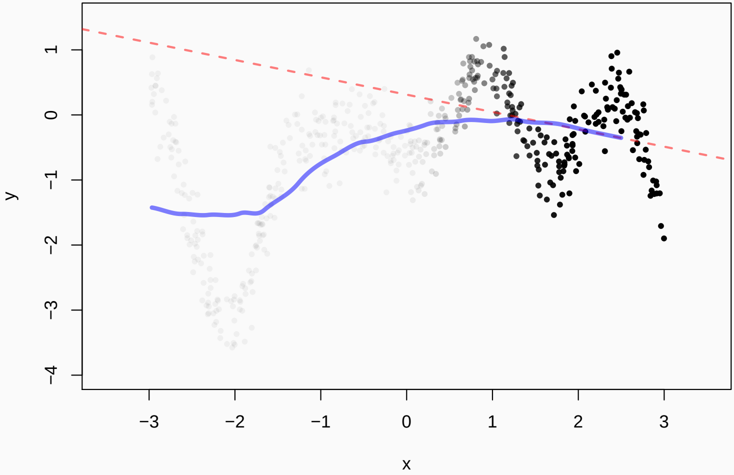


LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

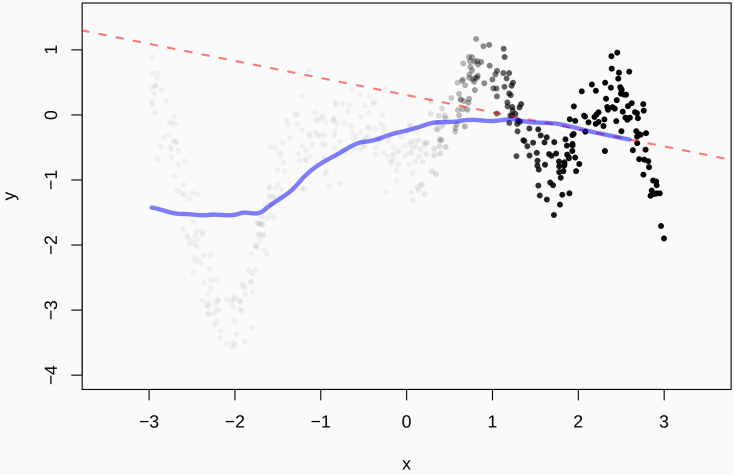
R:

```
l = loess(y~x, span=0.5, degree=1)
```



Model: (non-parametric)

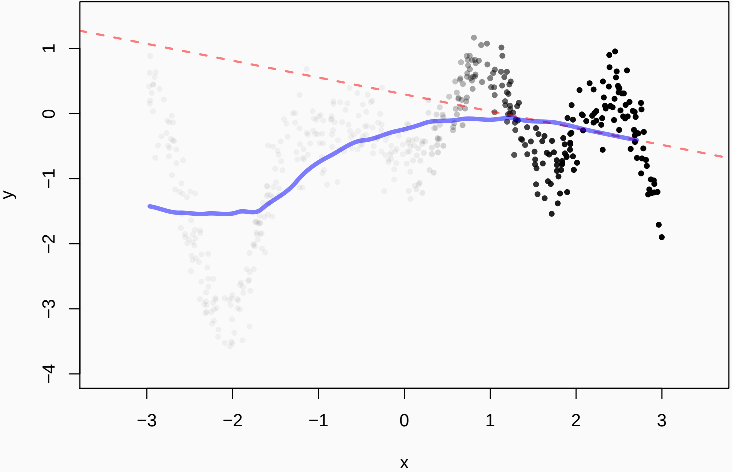
R:
```
l = loess(y~x, span=0.5, degree=1)
```



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

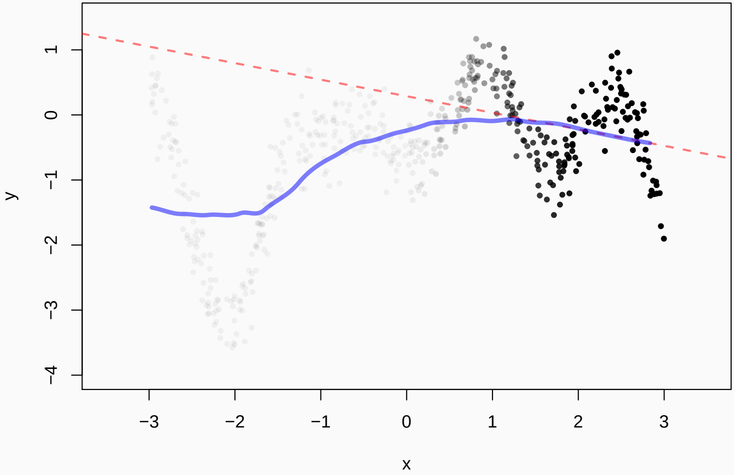
R:
```
l = loess(y~x, span=0.5, degree=1)
```



LOCAL REGRESSION - HOW DOES IT WORK?

Model: (non-parametric)

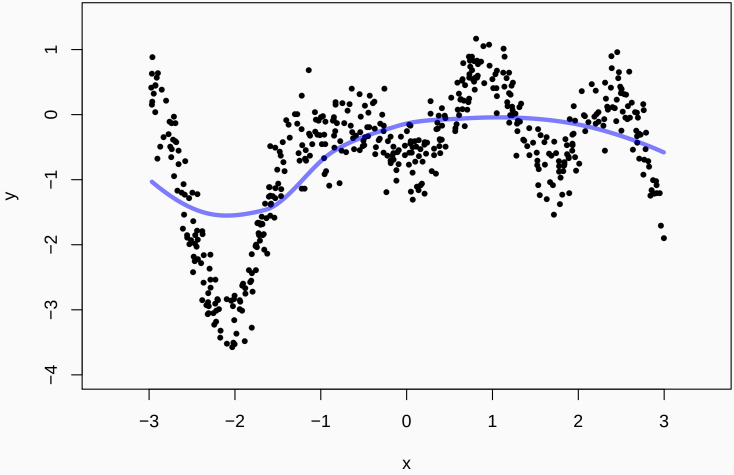
R: `l = loess(y~x, span=0.5, degree=1)`



LOCAL REGRESSION (LOESS) - ADJUSTING SPAN

Model: (non-parametric)

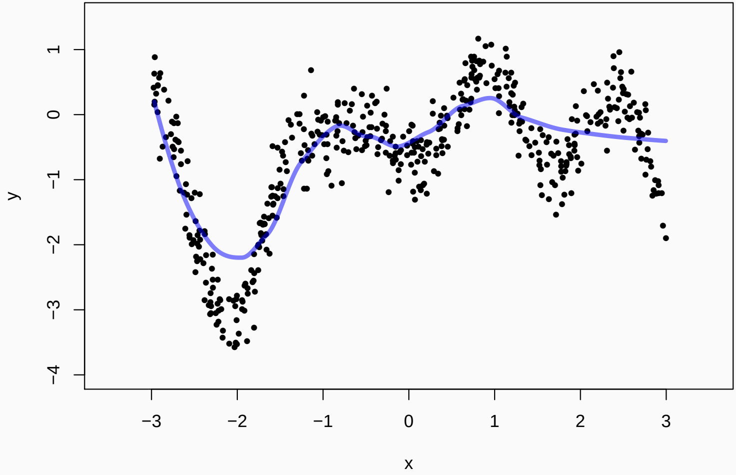
R: `l = loess(y~x, span=0.75)`



LOCAL REGRESSION (LOESS) - ADJUSTING SPAN

Model: (non-parametric)

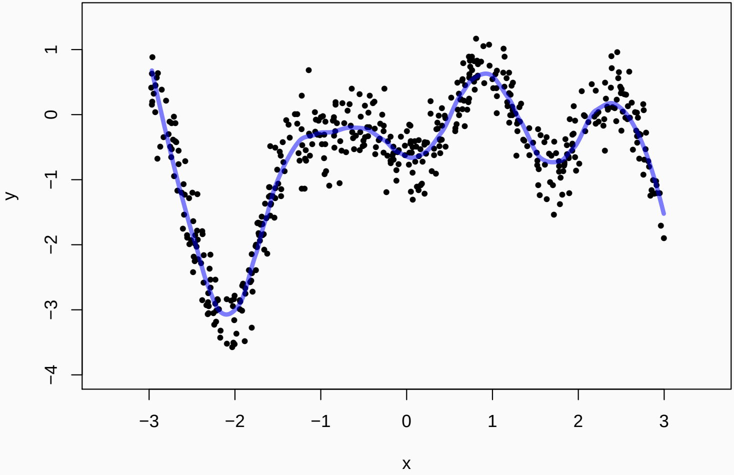
R: `l = loess(y~x, span=0.5)`



LOCAL REGRESSION (LOESS) - ADJUSTING SPAN

Model: (non-parametric)

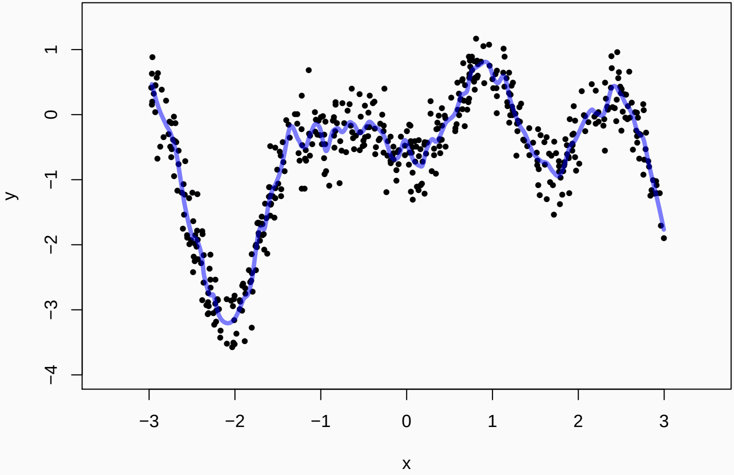
R: `l = loess(y~x, span=0.25)`



LOCAL REGRESSION (LOESS) - ADJUSTING SPAN

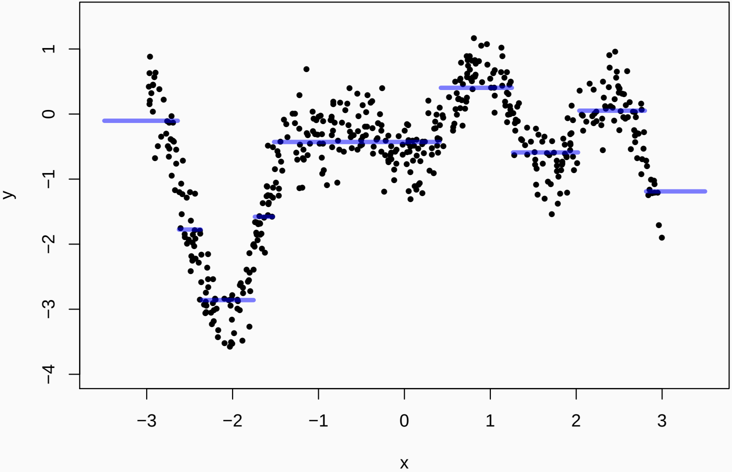
Model: (non-parametric)

R: `l = loess(y~x, span=0.05)`

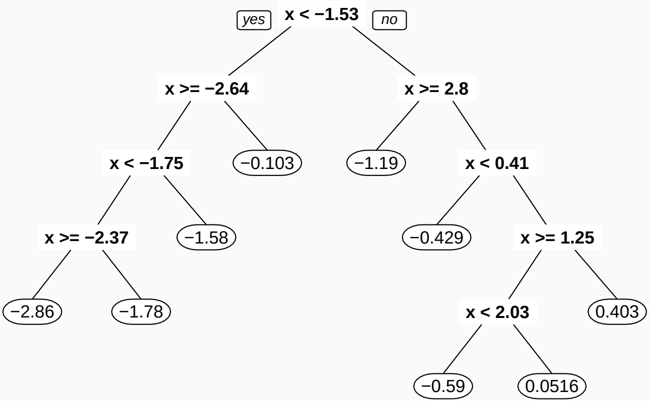


R:

```
l = rpart(y~x)
```

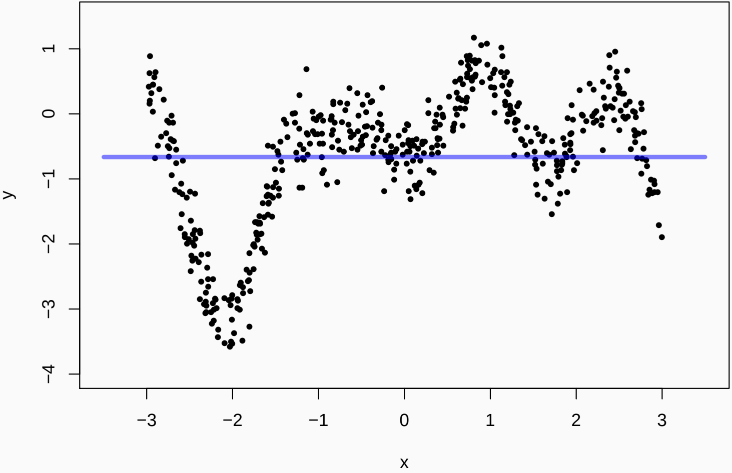


Model:

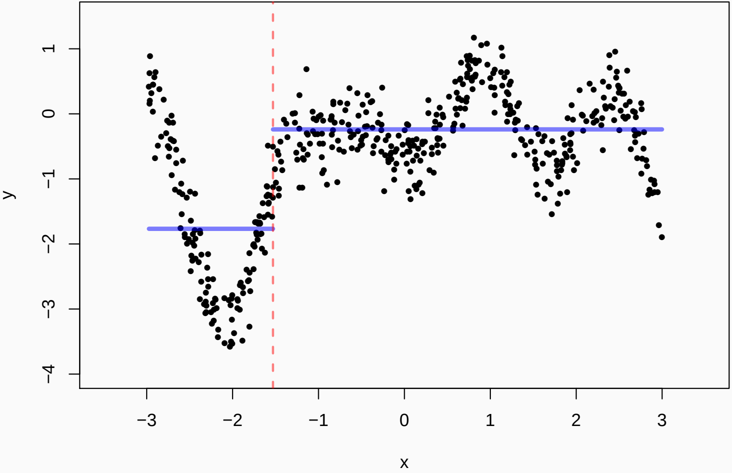


yes
x < −1.53
no
x >= −2.64
x >= 2.8
x < −1.75
−0.103
−1.19
x < 0.41
x >= −2.37
−1.58
−0.429
x >= 1.25
−2.86
−1.78
x < 2.03
0.403
−0.59
0.0516

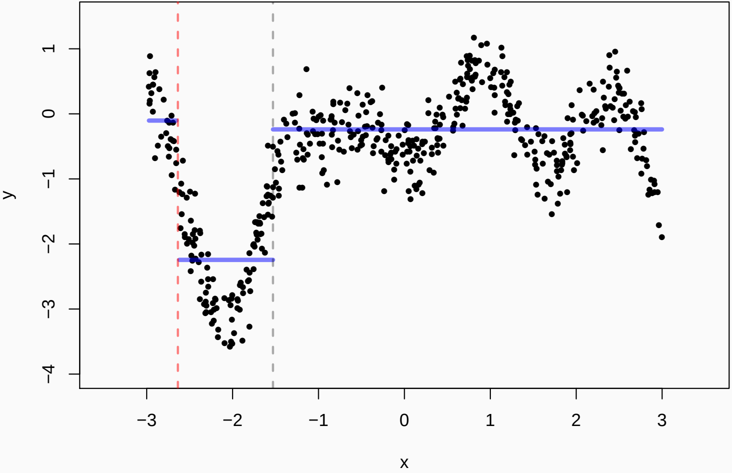
REGRESSION TREES - HOW DOES IT WORK?



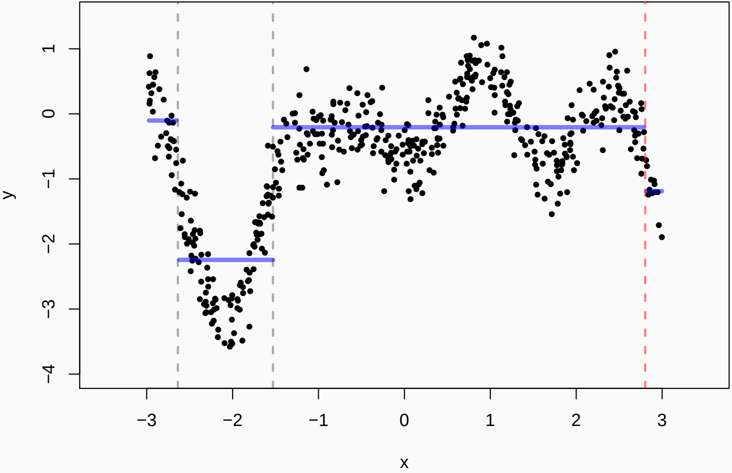
REGRESSION TREES - HOW DOES IT WORK?



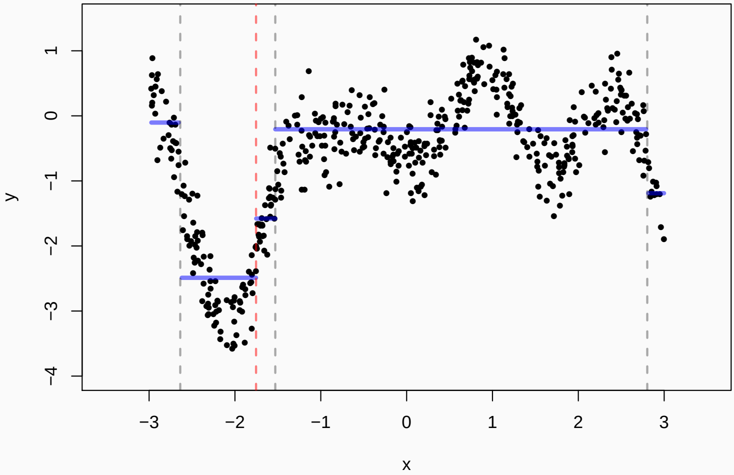
REGRESSION TREES - HOW DOES IT WORK?



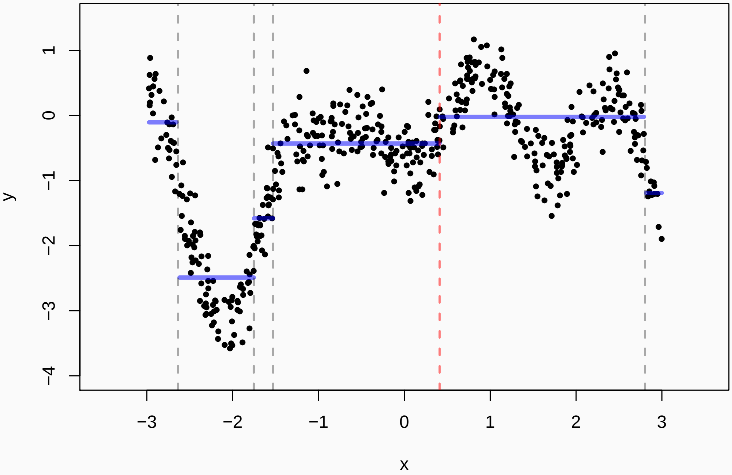
REGRESSION TREES - HOW DOES IT WORK?



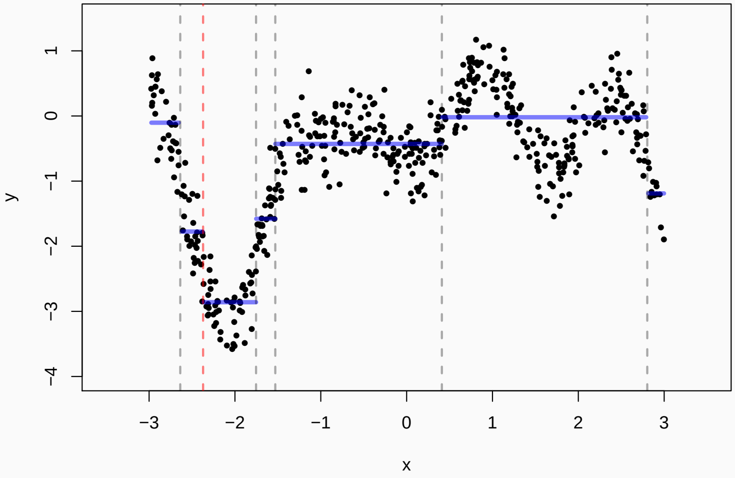
REGRESSION TREES - HOW DOES IT WORK?



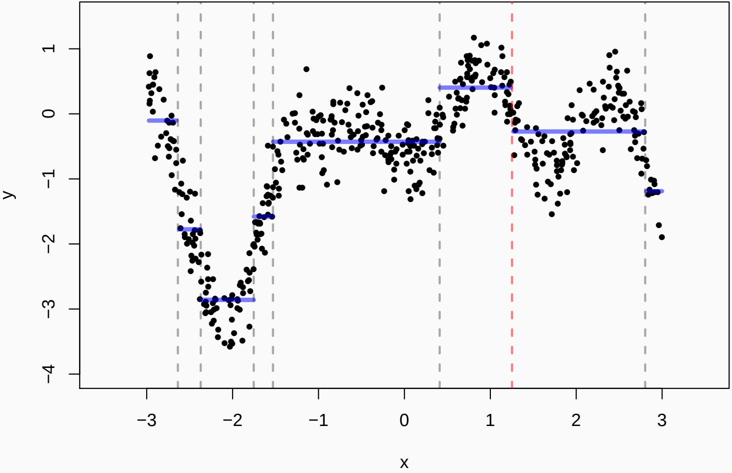
REGRESSION TREES - HOW DOES IT WORK?



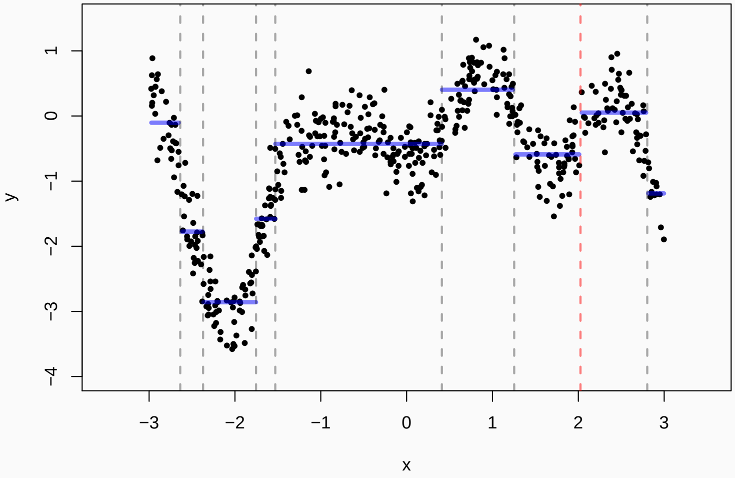
REGRESSION TREES - HOW DOES IT WORK?



REGRESSION TREES - HOW DOES IT WORK?



REGRESSION TREES - HOW DOES IT WORK?



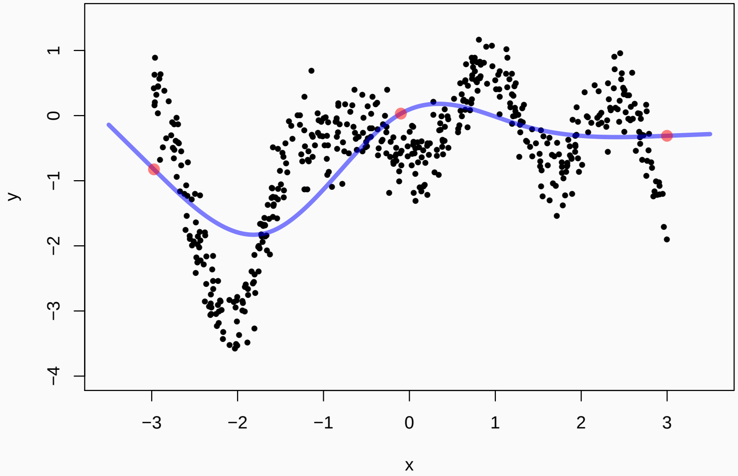
NATURAL SPLINE REGRESSION

Model:

$$y = \begin{cases} \beta_{0,1} + \beta_{1,1}\, x + \beta_{2,1}\, x^2 + \beta_{3,1}\, x^3 & \text{if } k_1 < x \le k_2 \\ \beta_{0,2} + \beta_{1,2}\, x + \beta_{2,2}\, x^2 + \beta_{3,2}\, x^3 & \text{if } k_2 < x \le k_3 \end{cases}$$

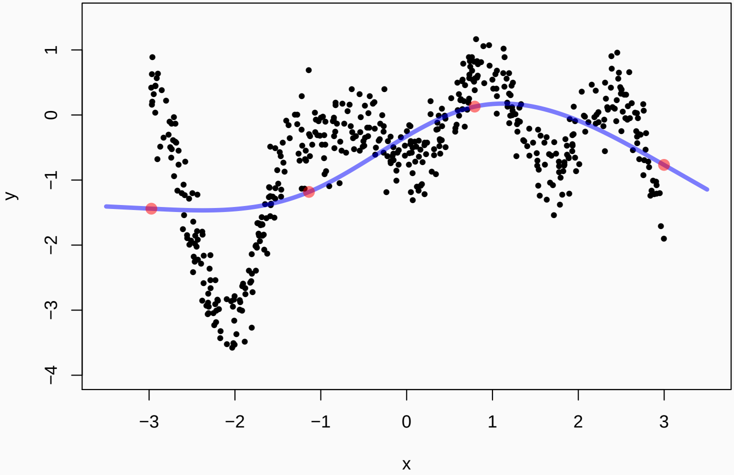
R:

```
l = lm(y~ns(x,df=4))
```



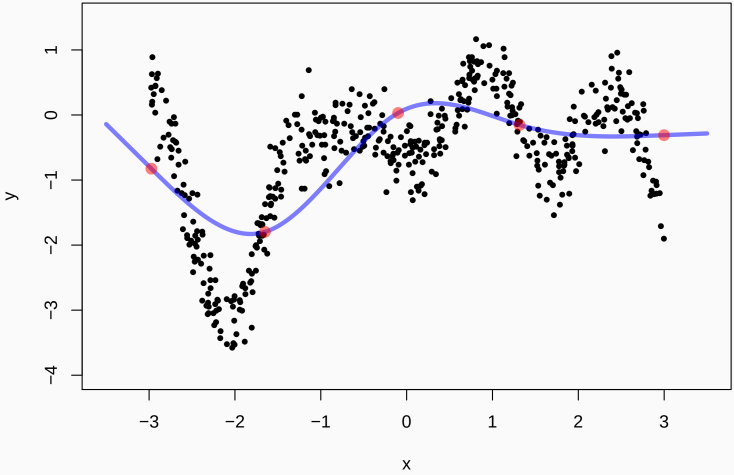
NATURAL SPLINE REGRESSION - ADDING KNOTS

R: `l = lm(y~ns(x,df=3))`



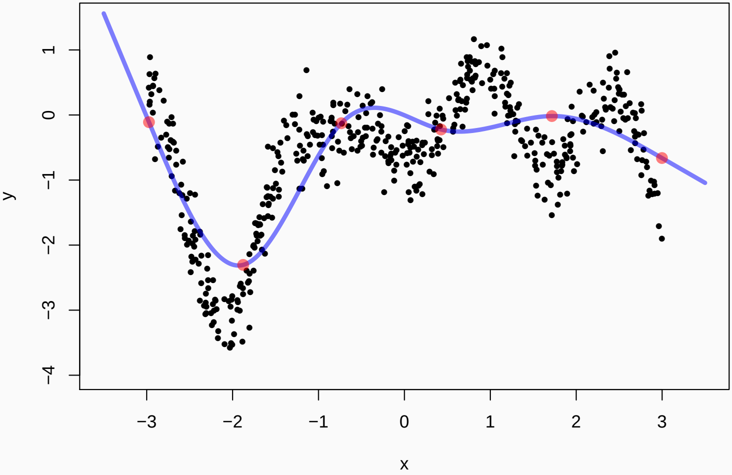
NATURAL SPLINE REGRESSION - ADDING KNOTS

R: `l = lm(y~ns(x,df=4))`



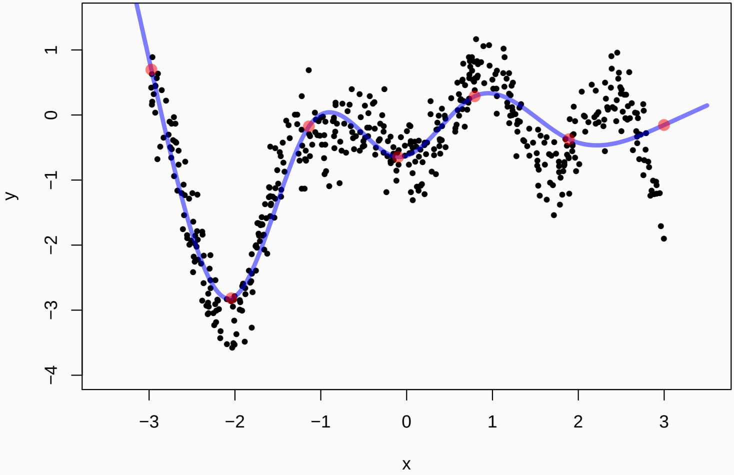
NATURAL SPLINE REGRESSION - ADDING KNOTS

R:
```
l = lm(y~ns(x,df=5))
```



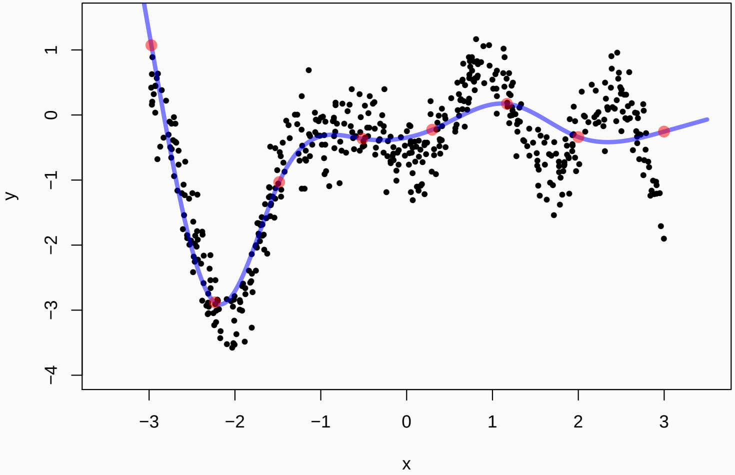
NATURAL SPLINE REGRESSION - ADDING KNOTS

R: `l = lm(y~ns(x,df=6))`



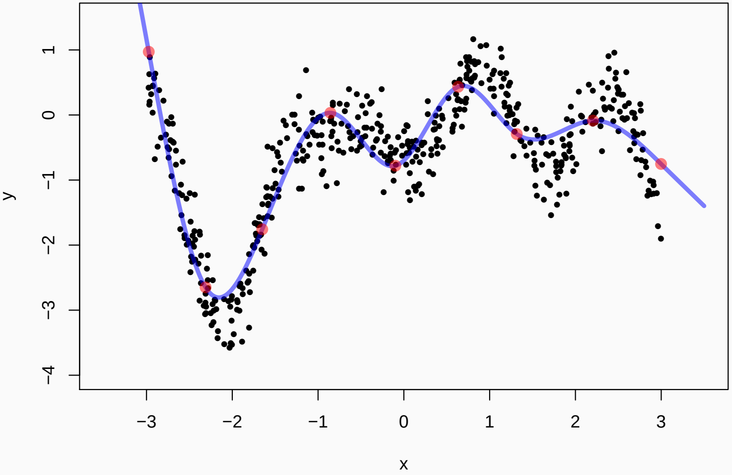
NATURAL SPLINE REGRESSION - ADDING KNOTS

R:
```
l = lm(y~ns(x,df=7))
```



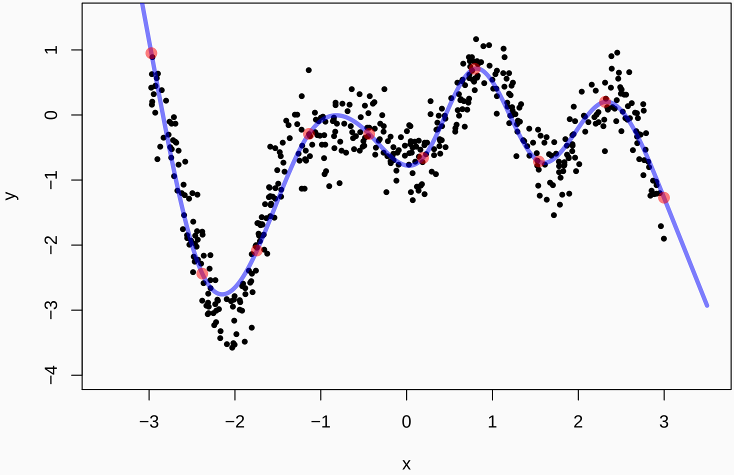
NATURAL SPLINE REGRESSION - ADDING KNOTS

R: `l = lm(y~ns(x,df=8))`



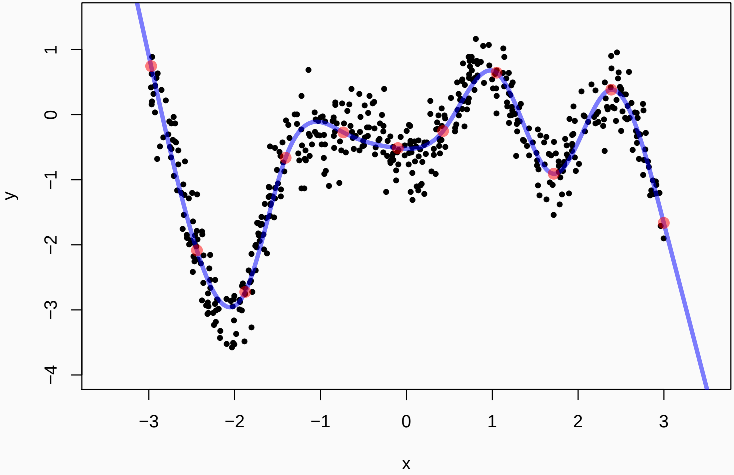
NATURAL SPLINE REGRESSION - ADDING KNOTS

R:
```
l = lm(y~ns(x,df=9))
```



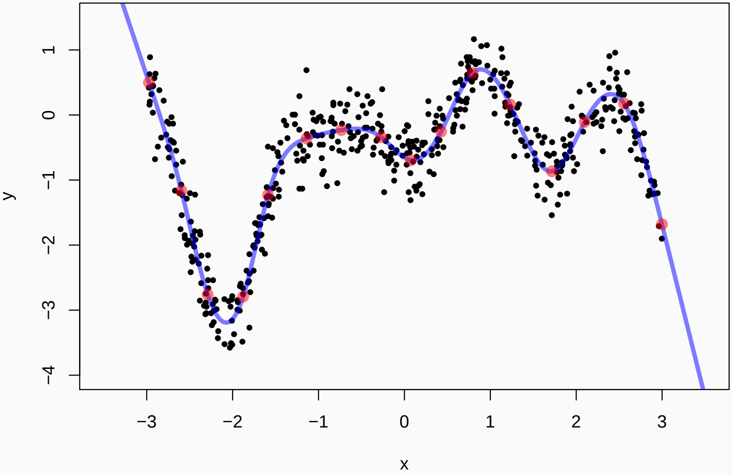
NATURAL SPLINE REGRESSION - ADDING KNOTS

R: `l = lm(y~ns(x,df=10))`



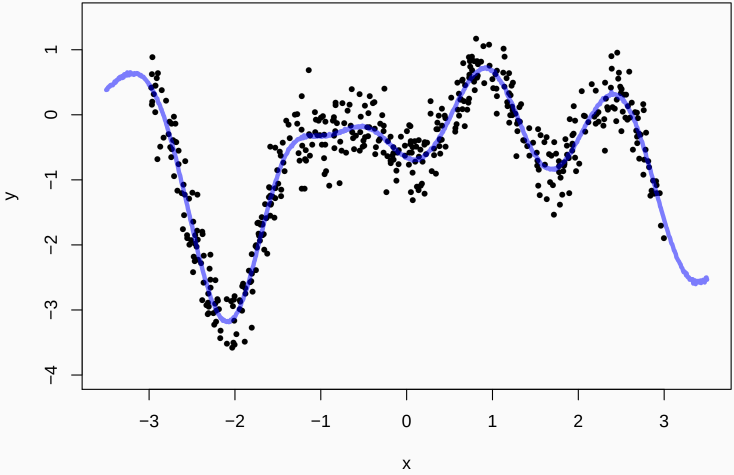
NATURAL SPLINE REGRESSION - ADDING KNOTS

R: `l = lm(y~ns(x,df=15))`



Model: $$y \sim \mathcal{N}(\beta_0, \Sigma)$$

R: `l = bgp(x, y, corr="exp")`



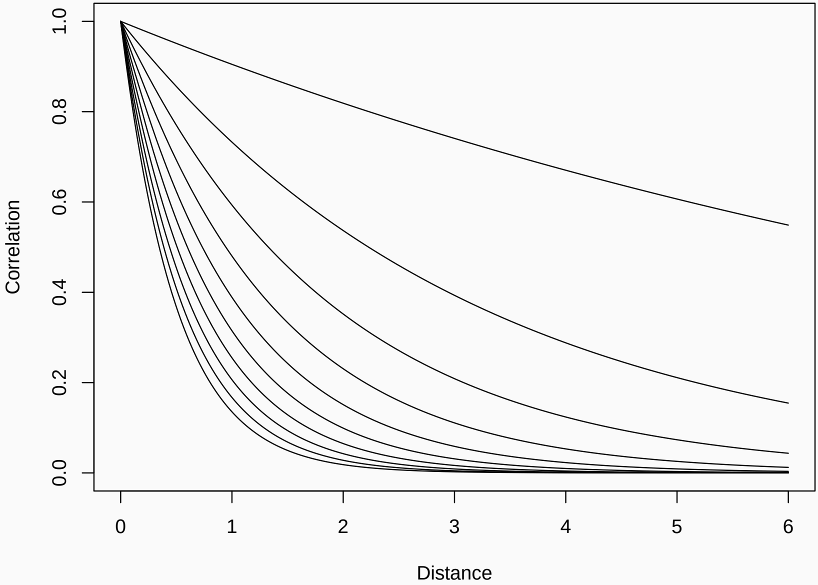
We model the outcome $\boldsymbol{y}$ as arising from a multivariate normal distribution given by

$$f(\boldsymbol{x}|\boldsymbol{\beta_0}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\beta_0})' \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\beta_0})\right).$$

The magic here comes from defining the covariance terms (off-diagonal elements) of $\boldsymbol{\Sigma}$ in a way such that close observations, in x, are more correlated.

$$\Sigma_{\{i,j\}} = \text{cov}(x_i, x_j) = \sigma^2 \, e^{-|x_i - x_j| \, \phi} \text{cov}(x_i, x_j) \qquad = e^{-|x_i - x_j| \, \phi}$$

EXPONENTIAL CORRELATION FUNCTION



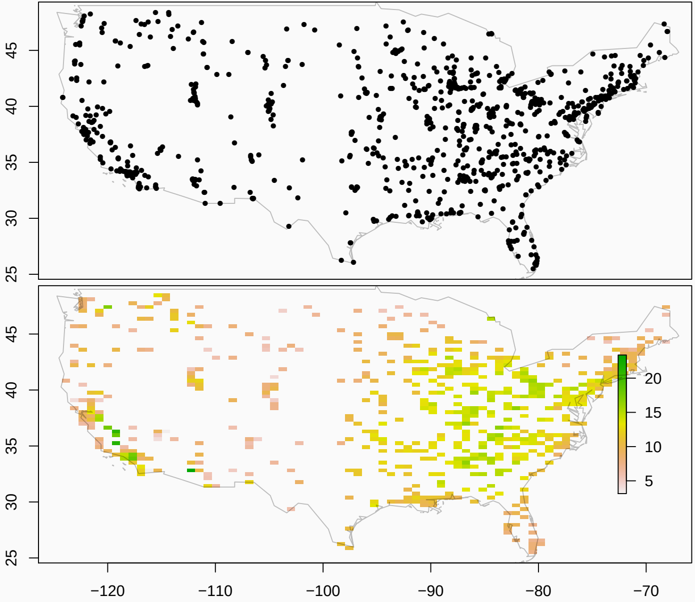
MOVING INTO SPACE

$PM_{2.5}$ BACKGROUND

Fine particulate matter ($PM_{2.5}$) is an EPA regulated air pollutant linked to a variety of adverse health effects

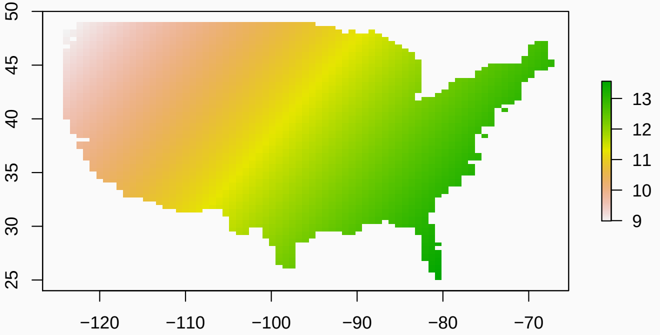
- Classified based on particle size ($< 2.5\ \mu$m diameter)
- Major species: Sulfate, Nitrate, Ammonium, Soil, Carbon.
- Minor species: trace elements (K, Mg, Ca), heavy metals (Cu, Fe), etc.
- Complex spatio-temporal dependence between species

SPATIAL DATA - TOTAL $PM_{2.5}$



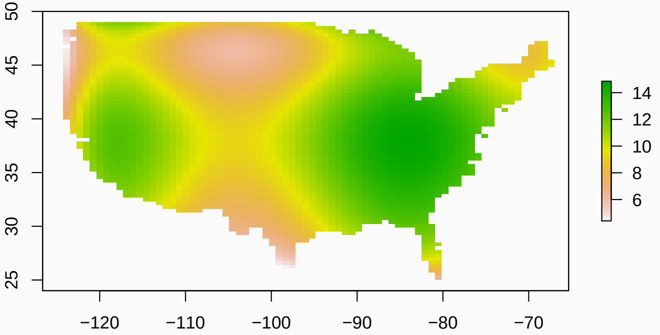
SPATIAL DATA - MULTIPLE REGRESSION

R: `l = lm(pm25~long+lat, data=d)`



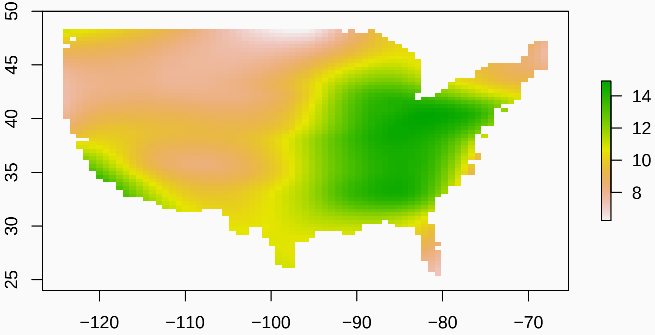
SPATIAL DATA - POLYNOMIAL REGRESSION

R: `l = lm(pm25~poly(long,5)+poly(lat,5), data=d)`



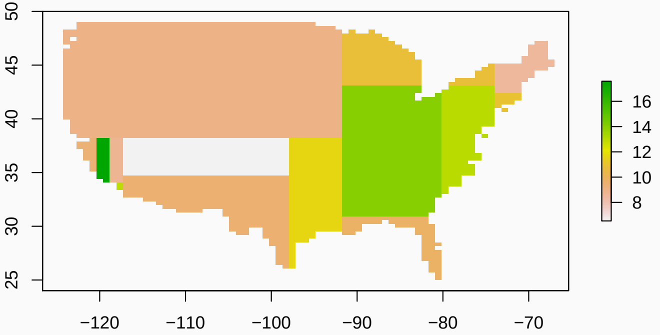
SPATIAL DATA - LOESS

R: `l = loess(pm25~long+lat, data=d, span=0.25)`



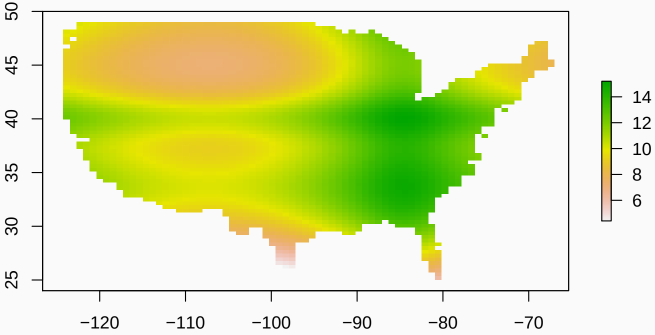
SPATIAL DATA - REGRESSION TREE

R: `l = rpart(pm25~long+lat, data=d)`



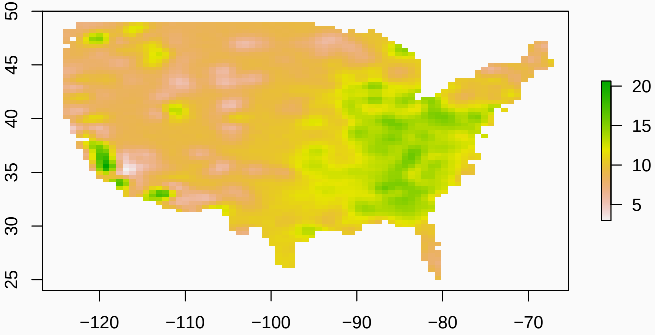
SPATIAL DATA - NATURAL SPLINES

R: `l = lm(pm25~ns(long,df=5)+ns(lat,df=5),data=d)`



SPATIAL DATA - GP

R: `l = bgp(X=locs, Z=pm25, XX=pred_locs, corr="exp")`



MODELING $PM_{2.5}$

ALL THE $PM_{2.5}$ DATA

Speciated $PM_{2.5}$ Sources

- Chemical Speciation Network (CSN) - 221 stations
- Interagency Monitoring of Protected Visual Environments (IMPROVE) - 172 stations

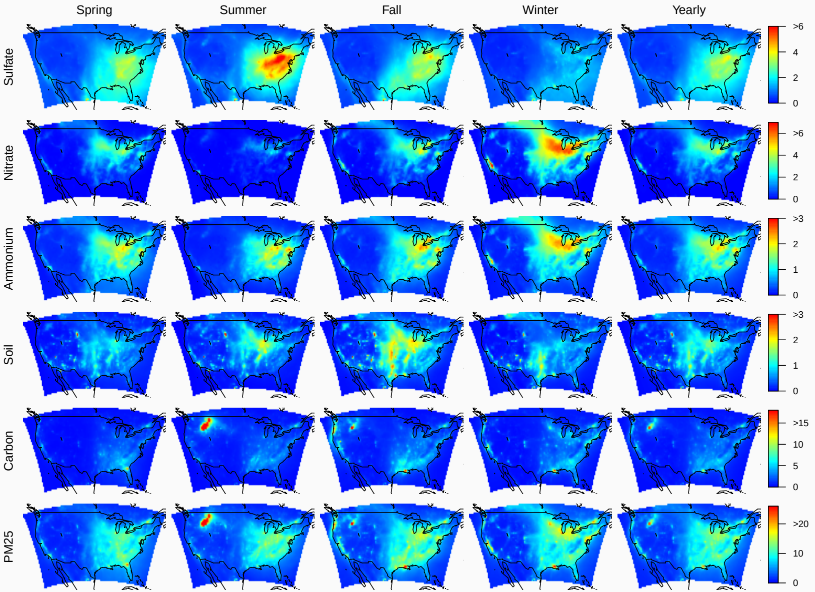
Total $PM_{2.5}$ Sources

- Federal Reference Method (FRM) - 949 stations

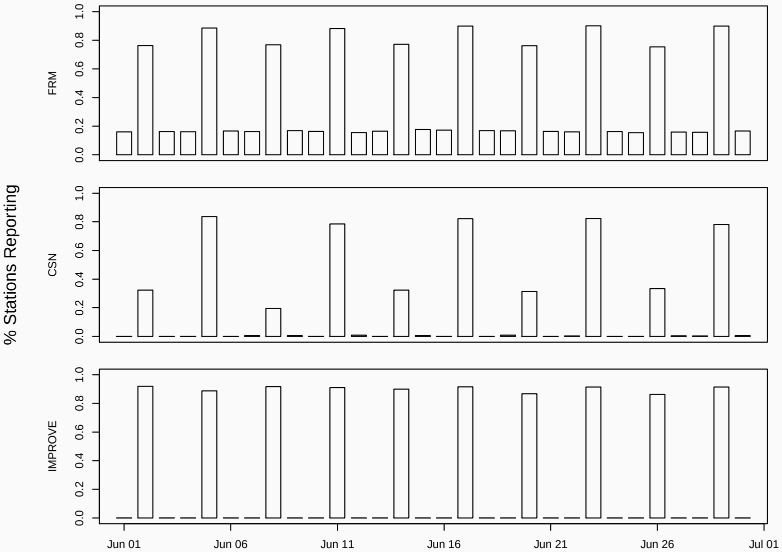
Model Output

- Community Multi-scale Air Quality (CMAQ) - 12 km grid

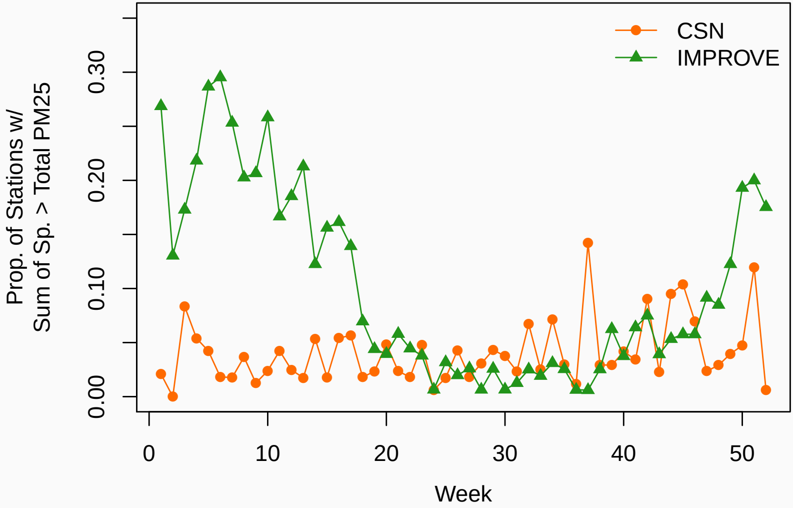
COMMUNITY MULTI-SCALE AIR QUALITY MODELING



DATA ISSUES - MONITORING FREQUENCY



DATA ISSUES - SUM VS TOTAL



Our goal was to model all 5 species and total $PM_{2.5}$ at the *same time*.

For each species level oberservation our model looked something like,

$$
\begin{aligned}
N_t^i(\mathbf{s}) &= Z_t^i(\mathbf{s}) + \epsilon_{N,t}^i(\mathbf{s}) \\
Z_t^i(\mathbf{s}) &= \max\left(0,\ \widetilde{Z}_t^i(\mathbf{s})\right) \\
\widetilde{Z}_t^i(\mathbf{s}) &= \beta_{0,t}^i + \beta_{0,t}^i(\mathbf{s}) + \beta_{1,t}^i\, Q_t^i(B_{\mathbf{s}})
\end{aligned}
$$

Here $\beta_{0,t}^i(\mathbf{s})$ is the Gaussian process that induces spatial dependence.

Everything gets coupled together via the total $PM_{2.5}$, which we require to be the sum of the individual species.

$$N_t^{tot}(s) = Z_t^{tot}(s) + \epsilon_{F,t}^{tot}(s)$$

$$Z_t^{tot}(s) = Z_t^{Sulf}(s) + Z_t^{Nit}(s) + Z_t^{Amm}(s) + Z_t^{Soil}(s) + Z_t^{Carb}(s) + Z_t^{Oth}(s)$$

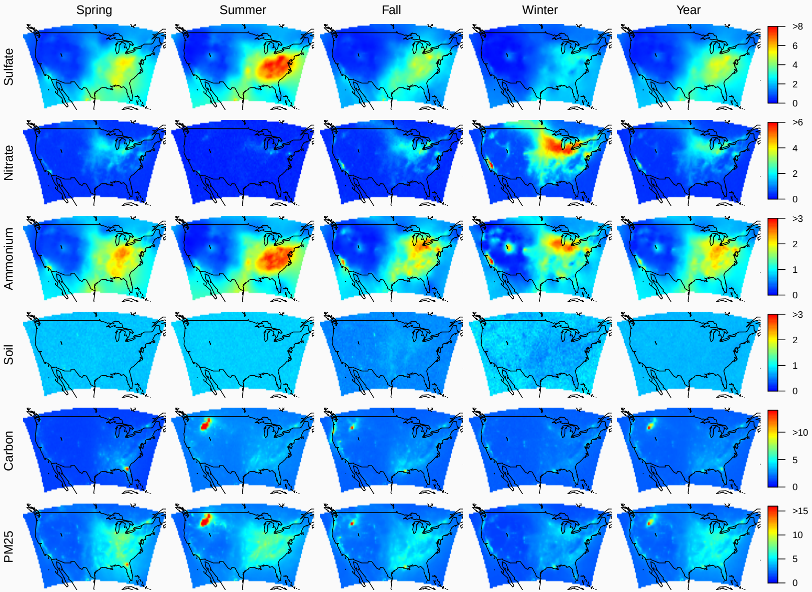
The wrinkle here is that we haven't observed all of the species (only the 5 major species) - so we end up having to estimate the "other" unobserved species as $Z_t^{Oth}(s)$.

COMPUTATION

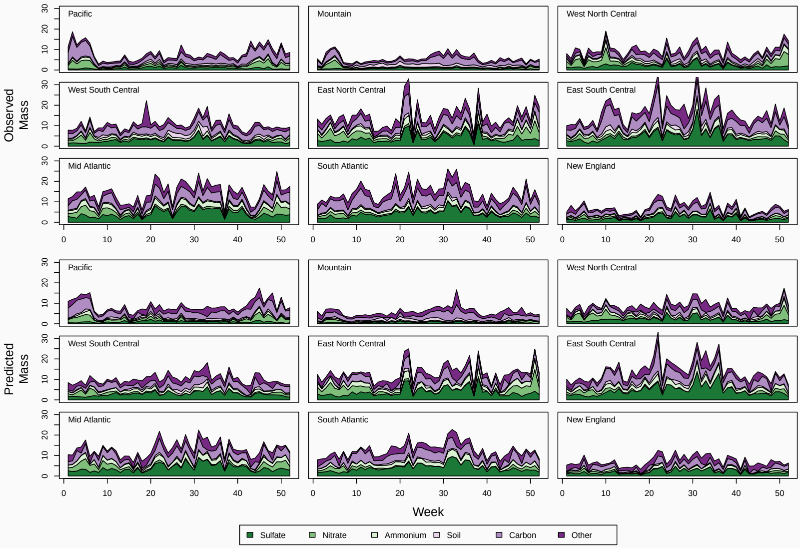
The model is fit and predictions made using a Bayesian hybrid MCMC based approach - which is very computationally intensive.

- Model fitting - $\sim$ 8 hours per weekly model
- Prediction - $\sim$ 7 hours per weekly model
- Yearly data - 52 weeks = $\sim$ 800 hours
- Model variants - 3 variant = $\sim 2,400$ hours
- Validation - 10-fold x-validation = $\sim 24,000$ hours

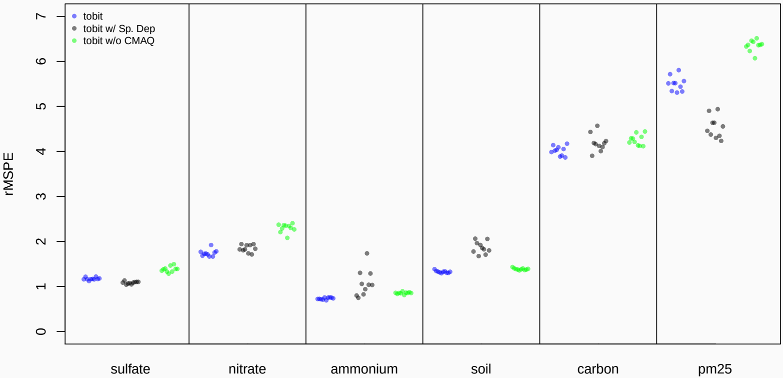
RESULTS - MAPS



RESULTS - TIME SERIES



VALIDATION



tobit
tobit w/ Sp. Dep
tobit w/o CMAQ
rMSPE
0
1
2
3
4
5
6
7
sulfate
nitrate
ammonium
soil
carbon
pm25

ACKNOWLEDGMENTS

ACKNOWLEDGMENTS

Paper - “A data fusion approach for spatial analysis of speciated PM2.5 across time” - Environmetrics

- Alan Gelfand - Duke
- Dave Holland - EPA
- Erin Schliep - Duke