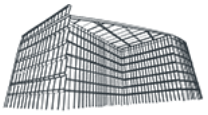
Topics in Machine Learning

Piecewise linear regression with splines

Ben Meuleman, Ph.D.
Swiss Center for Affective Sciences

R Lunch
December 3, 2019, Geneva

UNIVERSITÉ
DE GENÈVE



campus biotech

Contents

1. Background

Personal background

- 2003 – 2007 Bachelor in Psychology
- 2007 – 2009 Master in Psychology
- 2009 – 2010 Master in Statistical Data Analysis
- 2011 – 2015 Ph.D. in Psychology
- 2016 – 2019 Postdoc in Psychology
- 2019 – … Statistician

- Doctoral dissertation—including two publications—written on the application of machine learning to emotion data.

- Part-time statistical assistant at my psychology department (2011–2019) and contributor of course materials in artificial neural networks (University of Ghent, 2012–2017).

- R Lunch last year about machine learning, on “nearest neighbors”, “decision trees”, and “random forests”.

Background literature – Technical

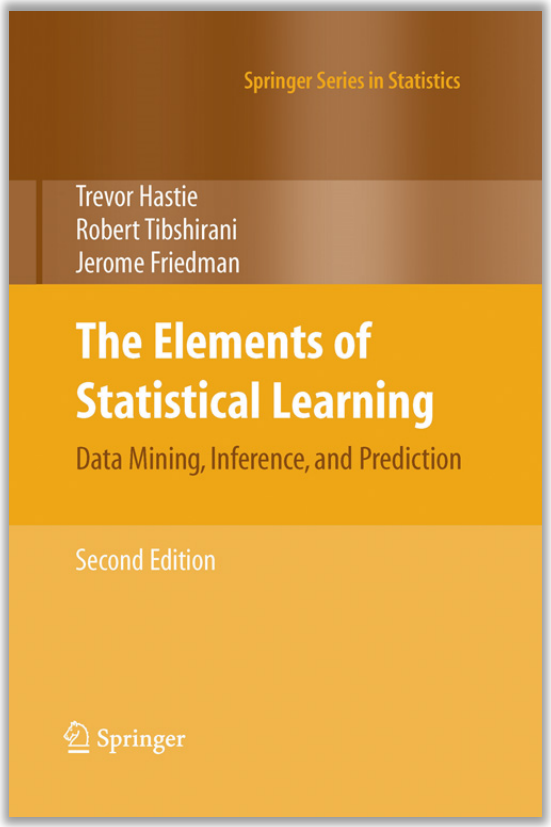
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer.

- Relatively accessible and comprehensive treatment of supervised and unsupervised methods for machine learning.

- Mathematically detailed but illustrated with practical data problems and many graphs.

- Other books on machine learning tend to be mathematically dense and overly focused on theory!

- Available online for free (via a university network)!



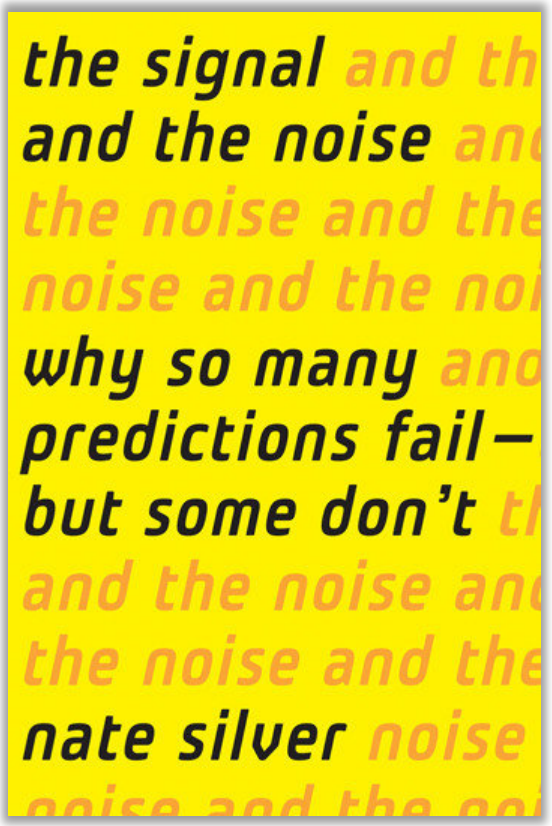
Background literature – Non-technical

- Silver, N. (2012). *The Signal and the Noise : Why So Many Predictions Fail – but Some Don't*. Penguin.

- Popular account of machine learning. Covers both recent successes and failures of machine learning.

- Chapters are organised according to specific prediction problems, e.g., weather forecasting, volcano eruption detection, sports prediction, chess, etc.

- Not mathematical but nevertheless offers an accessible introduction to many technical concepts (e.g., Bayesian inference).



Today's goals

- Today I will introduce to you some basic concepts about machine learning, as a primer. Following this, I will discuss a specific type of machine learner. At the end of this presentation you should be able to understand the following:

  1. That machine learning is just another type of statistical modelling

  2. That machine learning is connected to linear regression

  3. That we can introduce flexibility/nonlinearity in our basic regression model, without sacrificing interpretability

- The present material is adapted from a full 5-hour workshop on machine learning! We will not have time to go into many important ML topics today…

R part

- The R part of this R lunch will be light, as I prefer to focus on explaining the important concepts, rather than the somewhat boring details of how to run a model in R.

- However, I will discuss the "earth" package in detail, since it is a popular and well-documented package for piecewise linear regression.

- The code in these slides can be copied directly to your R script editor (all code in appendix). It requires the packages "splines", "earth", "plotmo", and "visreg" to be installed.

- Don't forget to check R's very comprehensive task views page on available packages for machine learning: https://cran.r-project.org/web/views/MachineLearning.html

2. Introduction

The hype…

DeepFake video manipulation
https://www.youtube.com/watch?v=VWrhRBb-1Ig

US & WORLD \ TECH \ ARTIFICIAL INTELLIGENCE

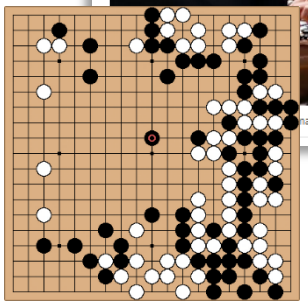
Former Go champion beaten by DeepMind retires after declaring AI invincible

'Even if I become the number one, there is an entity that cannot be defeated'

By James Vincent | Nov 27, 2019, 8:42am EST

SHARE

…atches with the AI program AlphaGo. | Photo: Google / Getty Images



Computer chess & Go

Self-driving vehicles

Machine learning as statistics

- In the media, machine learning is often presented as a kind of artificial intelligence, implemented in robots imitating human intelligence.

- This is a sensationalized version of machine learning, and may still take decades to achieve!

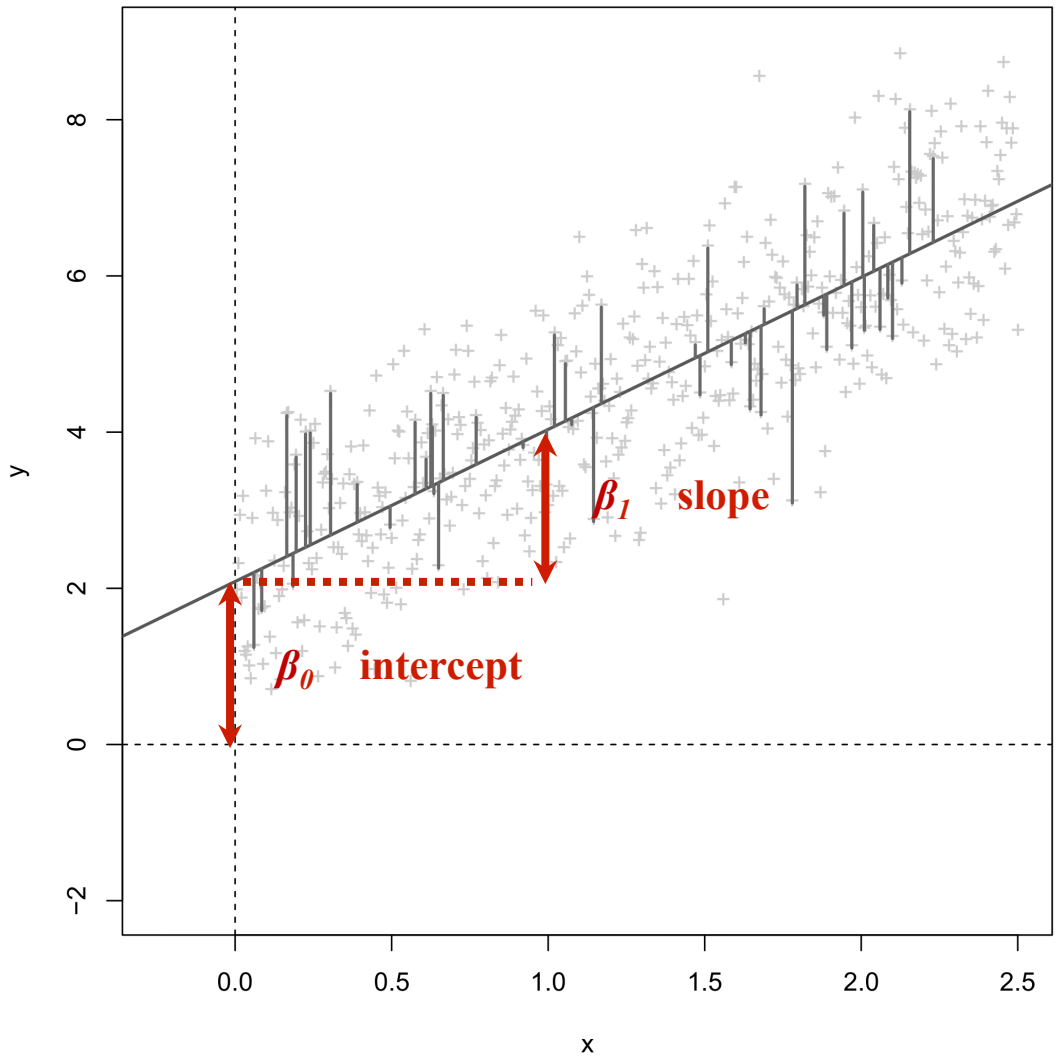
- More often, ML are just statistical algorithms that attempt to extract complex patterns from observed data sets (e.g., faces, voices). These algorithms are also referred to as "pattern recognition" or "data mining".

- Extracting data patterns with ML often involves the prediction of a certain dependent variable, given a number of predictors or independent variables. This is exactly the same as in linear regression…

(Hastie, Tibshirani & Friedman, 2005)

3. Machine learning and linear regression

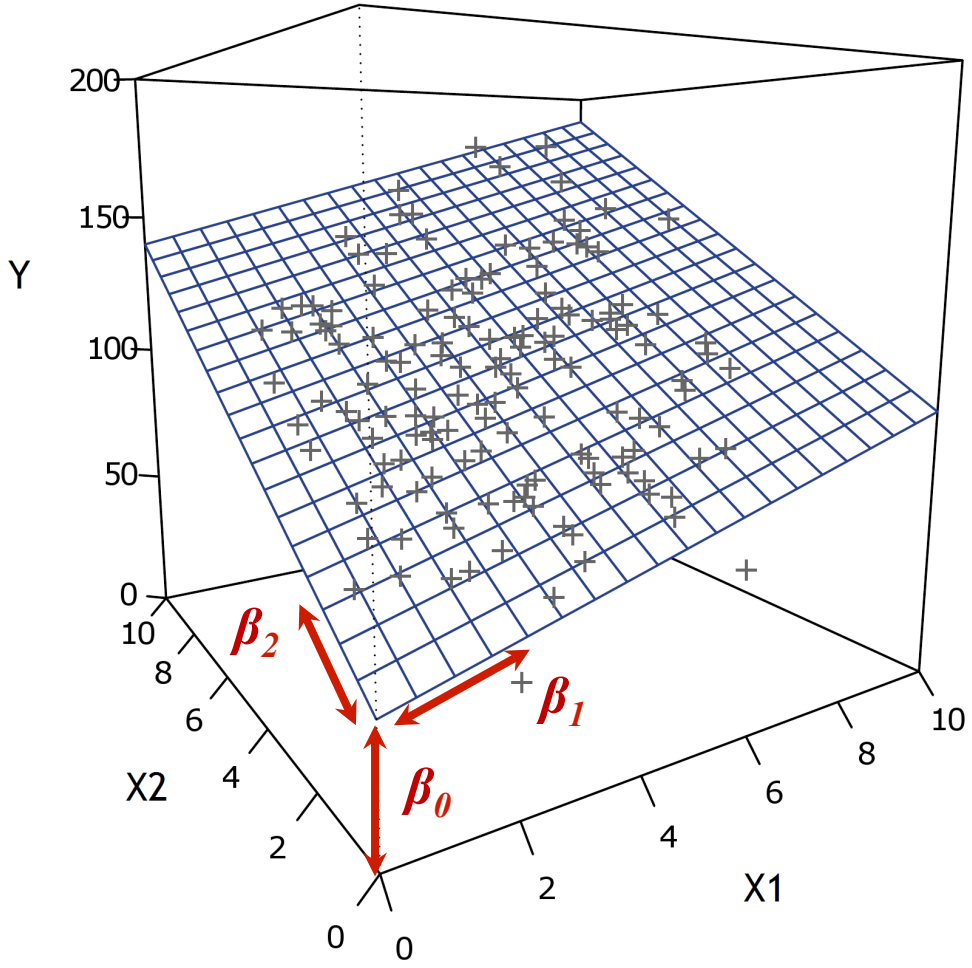
Linear regression



- Continuous dependent Y
- Continuous independent X
- $Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$

- Regression function minimizes the sum-of-the squared-distances to the line. This is a measure of error for the model

Linear regression



- $Y_i = \beta_0 + \beta_1 X_{i1} + \ldots + \beta_p X_{ip} + \varepsilon_i$

Linear regression

- When you analyze data, you are probably using some special version of the preceding model:

• One-way regression	One continuous independent
• Multiple regression	Multiple continuous independents
• Independent samples *t*-test	One categorical independent (2 levels)
• One-way ANOVA	One categorical independent (>2 levels)
• Multi-way ANOVA	Multiple categorical independents
• ANCOVA	Mix of categorical and continues independents
• Pearson correlation	One continuous independent. Dependent and independent standardized.
• One-sample *t*-test	Intercept-only as independent.
• Paired *t*-test	Difference score as dependent. Intercept-only as independent.

Regression as a machine learner

- Linear regression is itself a statistical learner and can be considered a machine learning model. Moreover, many advanced models for machine learning can be rewritten in regression language.

- In linear regression we often evaluate:

  A) The relevance/significance of IVs/predictors with *p*-values

  B) The model's fit to the dependent data with an R^2 value.

- These two goals have a more general counterpart in machine learning…

Same concept, different word…

Concept	Classic regression term		Machine learning term
Data	Observations		Cases
Y variables	Dependents		Responses / Targets / Outcomes
X variables	Independents / Predictors / Regressors		Features / Inputs
Parameters	Coefficients		Weights
Parameter estimation	Least squares estimation		Training / Learning
Model predictions	Fitted values		Predicted values
Relevance evaluation of X variables	Hypothesis testing	≈	Feature selection
Model fit	Proportion of explained variance (R^2)		Predictive strength (e.g., cross-validated R^2)

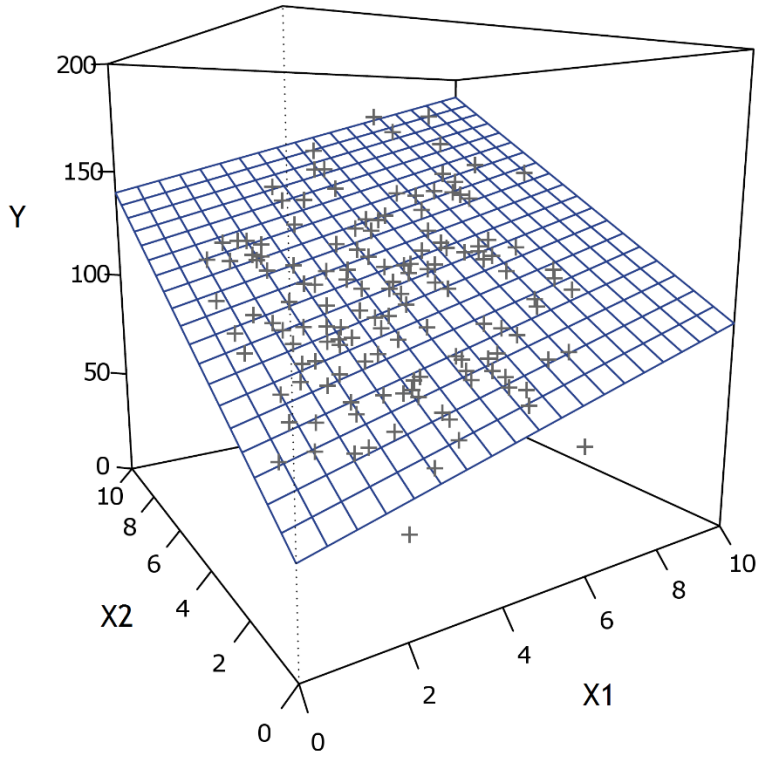
ML modelling problems

Response variable / Dependent (DV)	Predictors / Independents (IVs)
• Does a photographed face contain a smile, yes or no?	Images of faces, possibly decomposed to individual pixels
• Next speed and direction for a self-driving car	Current and past speed and direction, detected features of the road ahead, detected obstacles, etc.
• Probability of an e-mail being spam or not spam?	Message title, contents, sender, punctuation, spelling, grammar, etc.
• Should a patient be diagnosed with a tumor (which type)?	Medical images, observational data (e.g., test performance), demographical characeristics.
• Probability of death following lung injury, death vs survival?	Variables related to the injury, diagnostics, radiography, demographical characteristics
• Probability of experiencing a certain emotion (e.g., joy, fear, anger)	Experimental conditions, physiological markers, behavioural data, self-reports, etc.

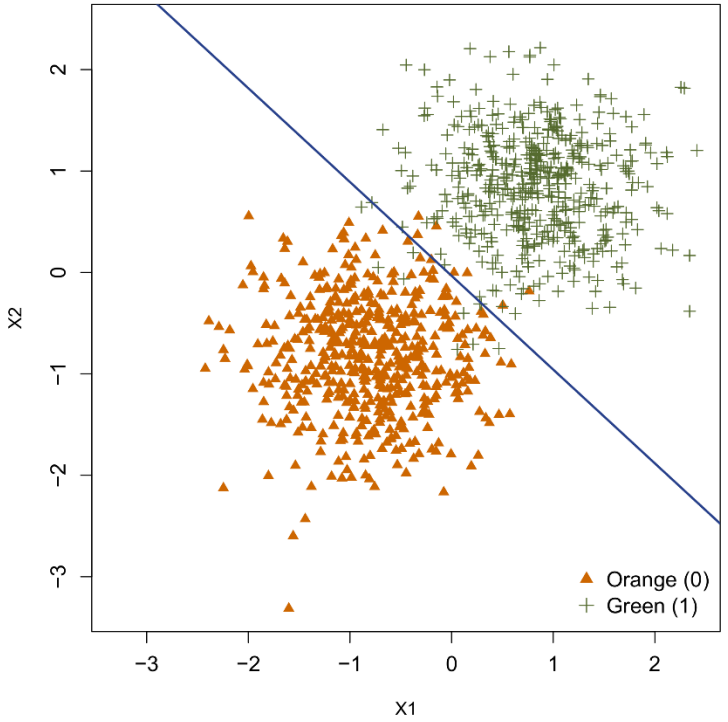
4. Why do we need ML?

Linear models

Continous dependent
"regression model"



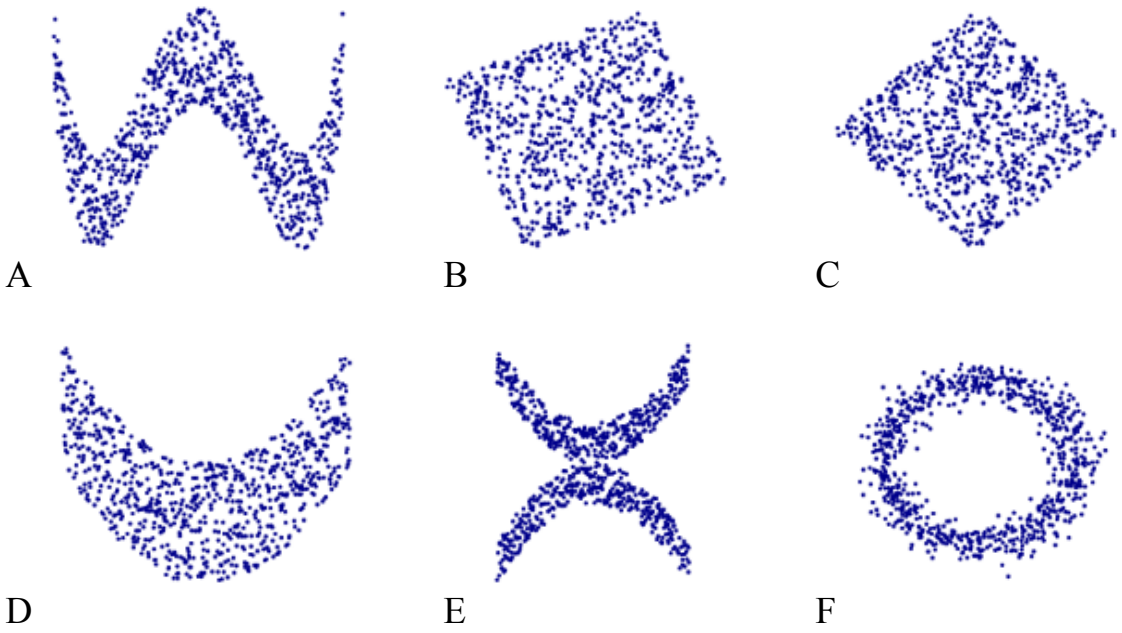
Categorical dependent
"classification model"



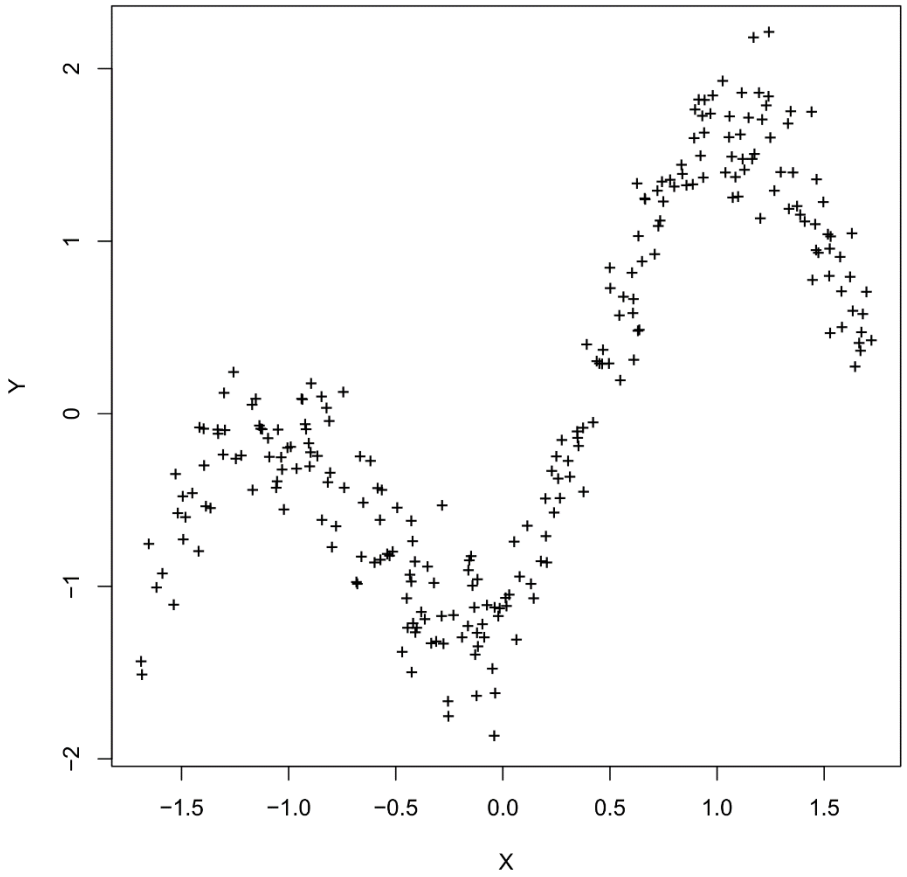
- Basic linear models have generally been very successful at modelling data, even when they are only approximations.

Nonlinearity problem

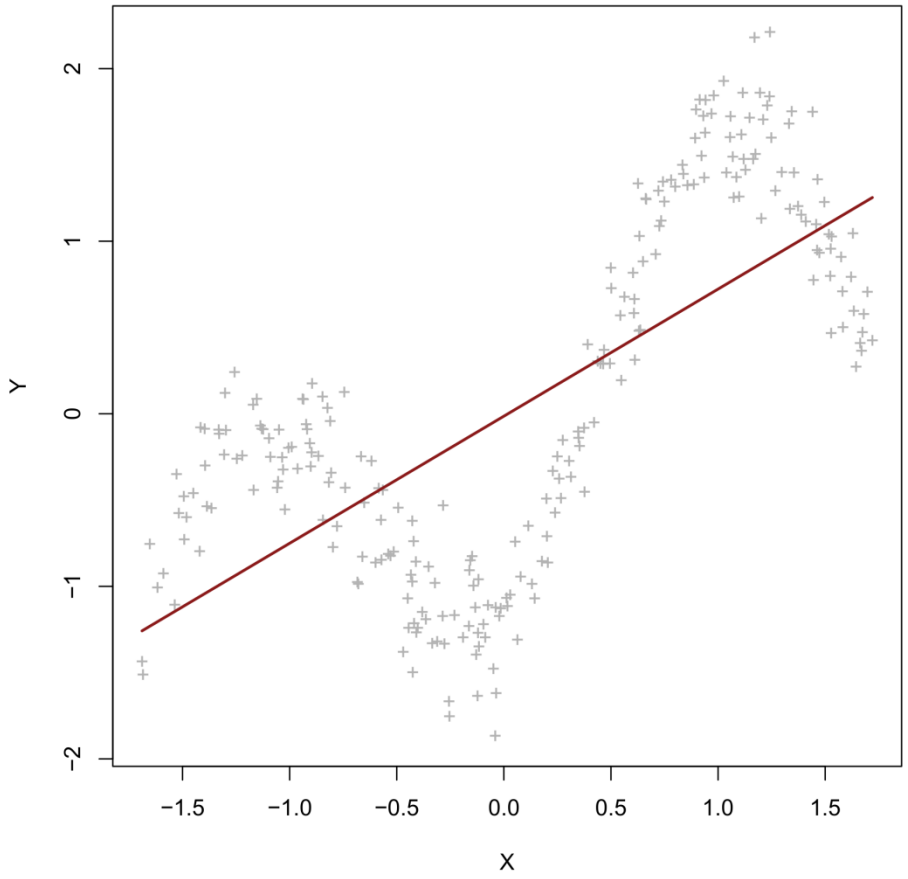
- However, nonlinear patterns can include complex curvilinear associations and/or interactions between multiple predictors simultaneously. Modelling nonlinearity is the primary reason why most ML models exist!



How can we achieve nonlinearity?



How can we achieve nonlinearity?

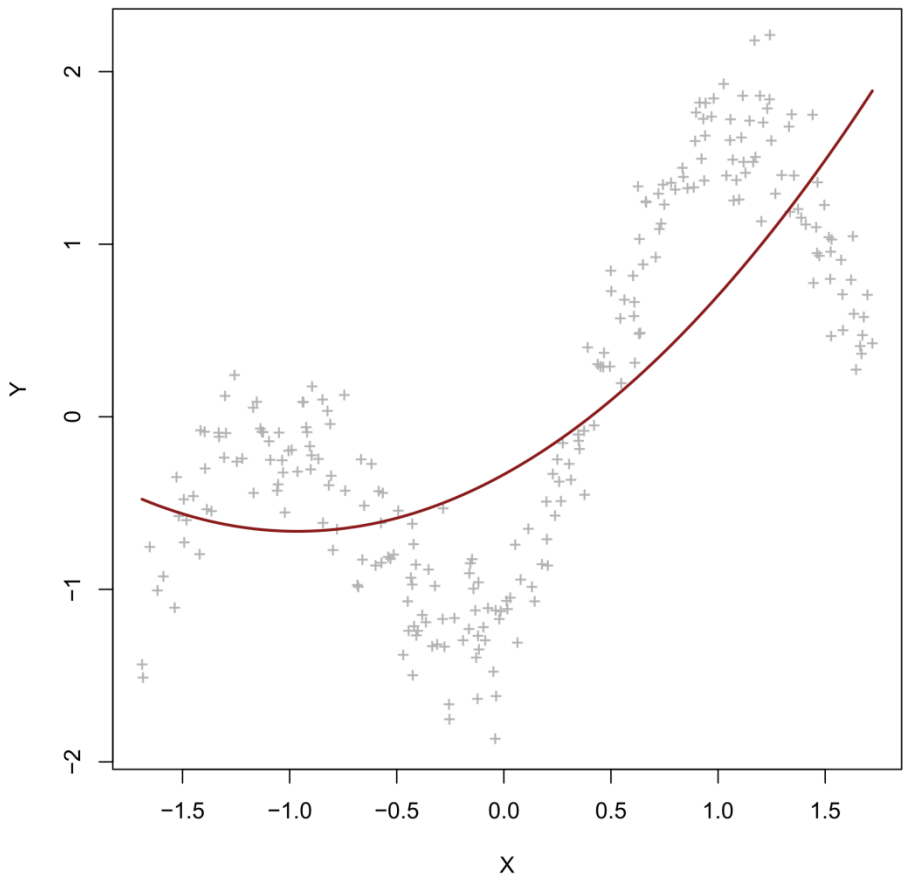


$$E(Y) = \beta_0 + \beta_1 X$$

X
-1.52
-1.23
-0.88
-0.57
0.01
0.39
0.55
1.01
...

Dimension of the predictor set: 1

How can we achieve nonlinearity?

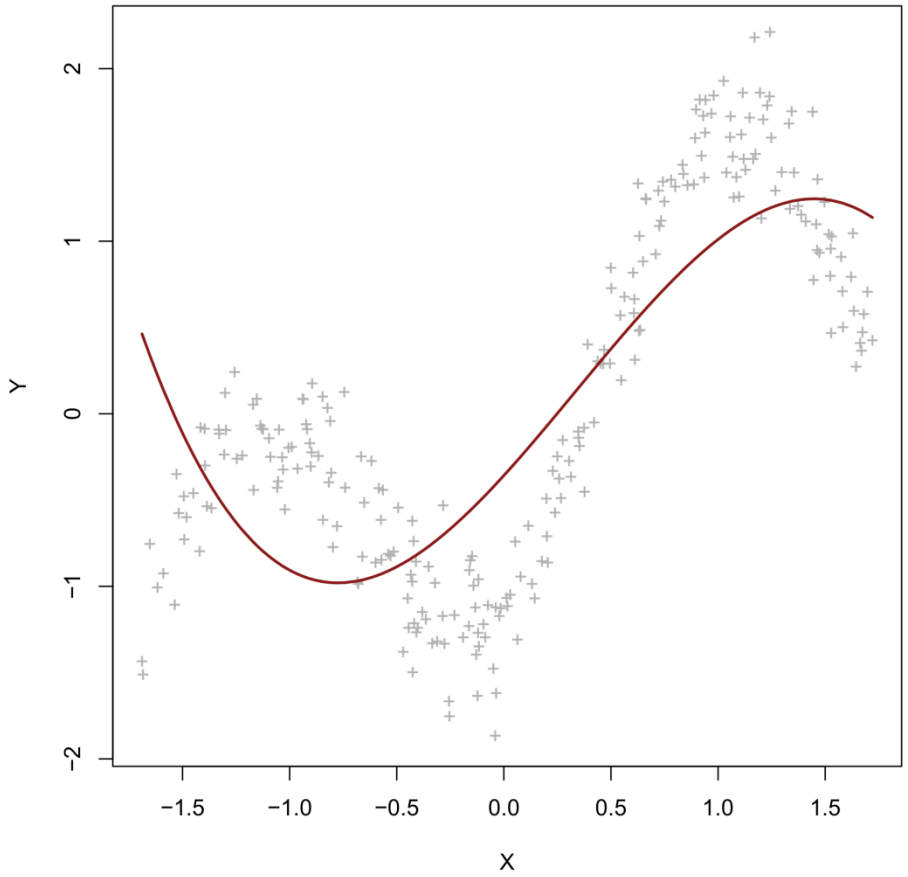


$$E(Y) = \beta_0 + \beta_1 X + \beta_2 X^2$$

X	X^2
-1.52	2.31
-1.23	1.51
-0.88	0.77
-0.57	0.32
0.01	0.00
0.39	0.15
0.55	0.30
1.01	1.02
...	...

Dimension of the predictor set: 2

How can we achieve nonlinearity?

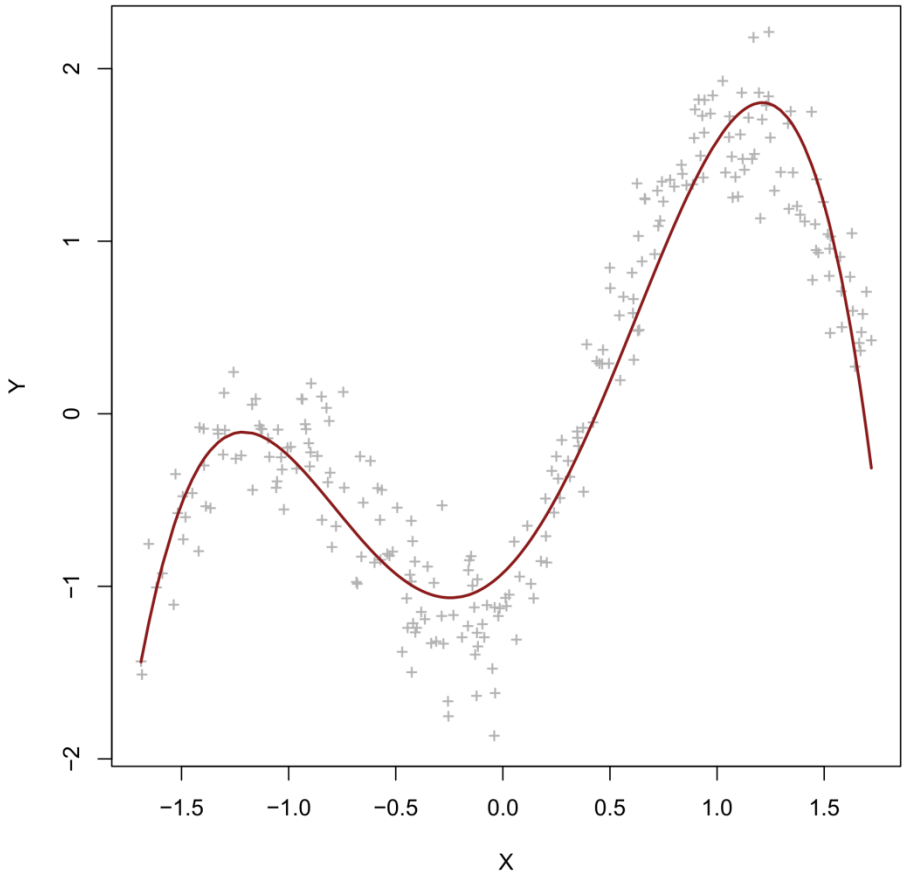


$$E(Y) = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3$$

X	X^2	X^3
-1.52	2.31	-3.51
-1.23	1.51	-1.86
-0.88	0.77	-0.68
-0.57	0.32	-0.18
0.01	0.00	0.00
0.39	0.15	0.05
0.55	0.30	0.16
1.01	1.02	1.03
...	...	...

Dimension of the predictor set: 3

How can we achieve nonlinearity?



$$E(Y) = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 X^4$$

X	X^2	X^3	X^4
-1.52	2.31	-3.51	5.33
-1.23	1.51	-1.86	2.28
-0.88	0.77	-0.68	0.59
-0.57	0.32	-0.18	0.10
0.01	0.00	0.00	0.00
0.39	0.15	0.05	0.02
0.55	0.30	0.16	0.09
1.01	1.02	1.03	1.04
…	…	…	…

Dimension of the predictor set: 4

Nonlinear predictor transformations

- Many models of machine learning operate in the manner just shown:

    1. First the model applies a clever nonlinear transformation to the independent/ predictor variables
    2. Then the model fits a basic linear model to the transformed data

- The preceding example shows a model that is linear in the parameters, but nonlinear in the predictors. The transformation used was a polynomial expansion of degree 4. Other than that, our model was not so special!

- A model can also be nonlinear in the parameters, but this type is a minority in machine learning (the most famous example being artificial neural networks).

Regression in R

- The following R code generates and plots the curvilinear data.

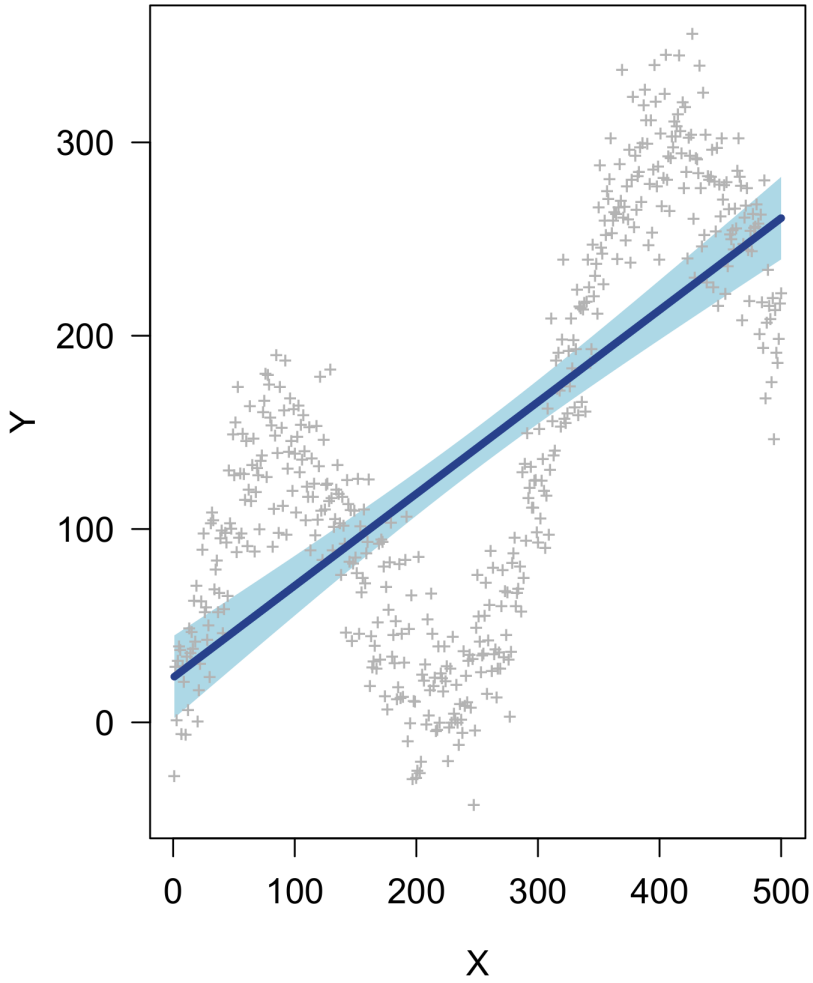
```
set.seed(668)
x <- 1:500
true <- 100*sin(0.02*x) + 0.5*x
y <- true + rnorm(500,0,25)
plot(x,y,pch="+",col="grey70",xlab="X",ylab="Y")
```

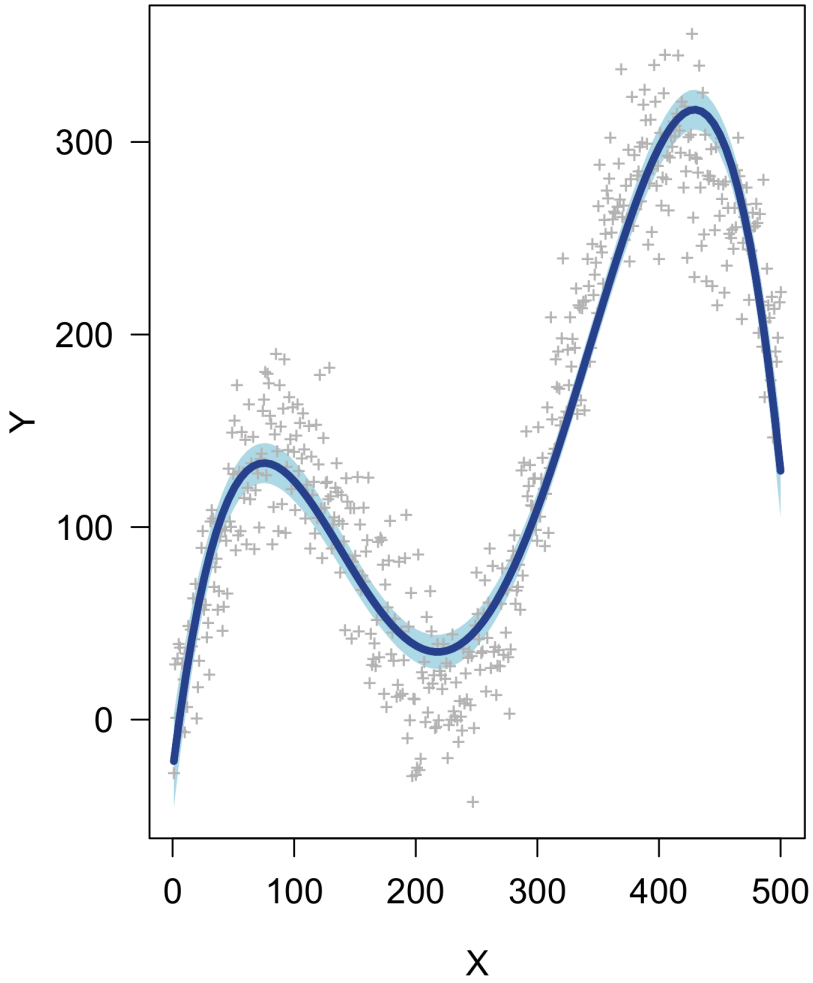
- We can fit basic linear and polynomial models as follows:

```
linear <- lm(y~x)
poly4 <- lm(y~poly(x,degree=4))
poly4 <- lm(y~x+I(x^2)+I(x^3)+I(x^4))      #alternative formula

visreg(linear,xvar="x")                    #package visreg!
visreg(poly4,xvar="x")
```

Regression in R





Regression in R

- More information about these models can obtained with the `summary` function, for example, for the `poly4` object:

```
Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)            142.235      1.473   96.57   <2e-16 ***
poly(x, degree = 4)1  1534.014     32.935   46.58   <2e-16 ***
poly(x, degree = 4)2   696.346     32.935   21.14   <2e-16 ***
poly(x, degree = 4)3  -369.581     32.935  -11.22   <2e-16 ***
poly(x, degree = 4)4 -1198.522     32.935  -36.39   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 32.94 on 495 degrees of freedom
Multiple R-squared:  0.8915,    Adjusted R-squared:  0.8906
F-statistic:  1017 on 4 and 495 DF,  p-value: < 2.2e-16
```

Problems with nonlinear transformations

- The preceding example makes nonlinear modelling seem relatively easy, but immediately raises important practical questions. For example, how do we decide…

  1. Which nonlinear transformation to apply?
     (e.g., logarithmic, exponential, polynomial,…)

  2. When to stop transforming?
     (e.g., at the 4^{th}, 10^{th}, 100^{th} degree polynomial)

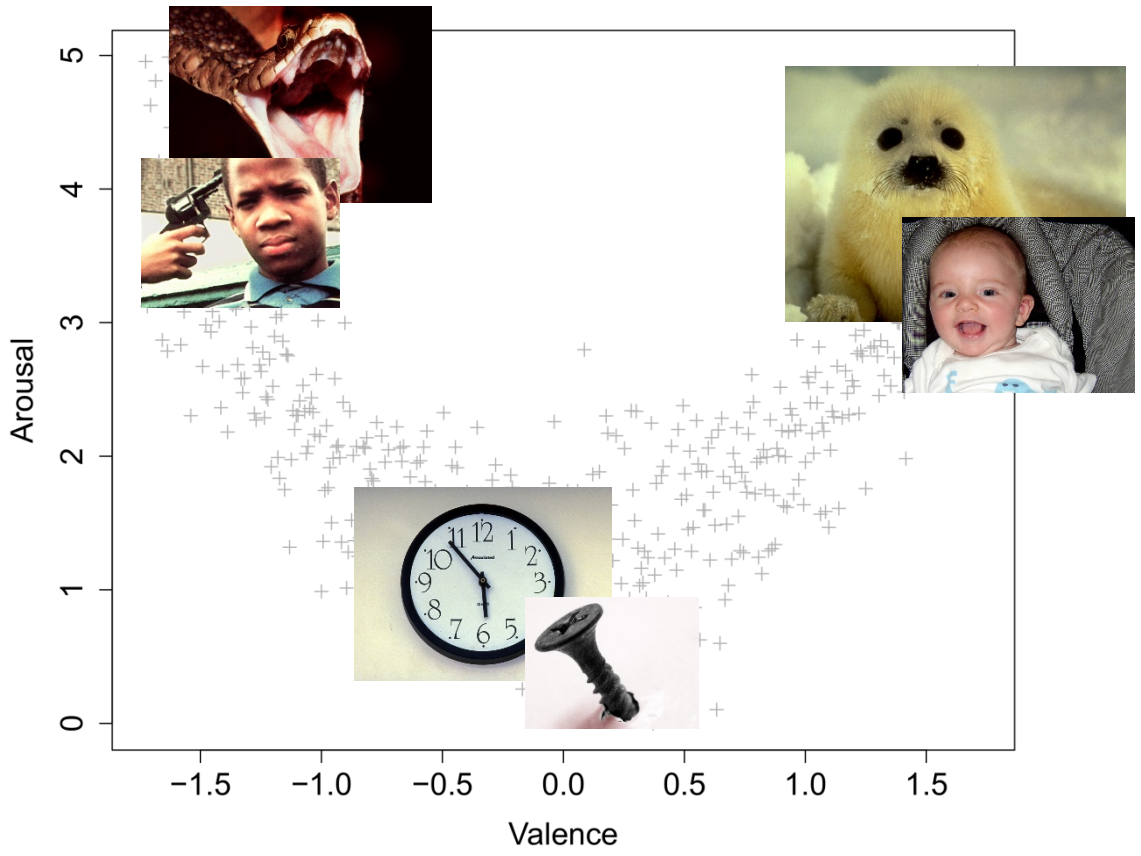
  3. How to interpret our nonlinear effects?

- Machine learning has generally sought to address all these issues with various models. For issues (1) and (3), however, there is one particular "trick" that can make life drastically easier…

5. Piecewise linear regression

Dealing with curves – A problem

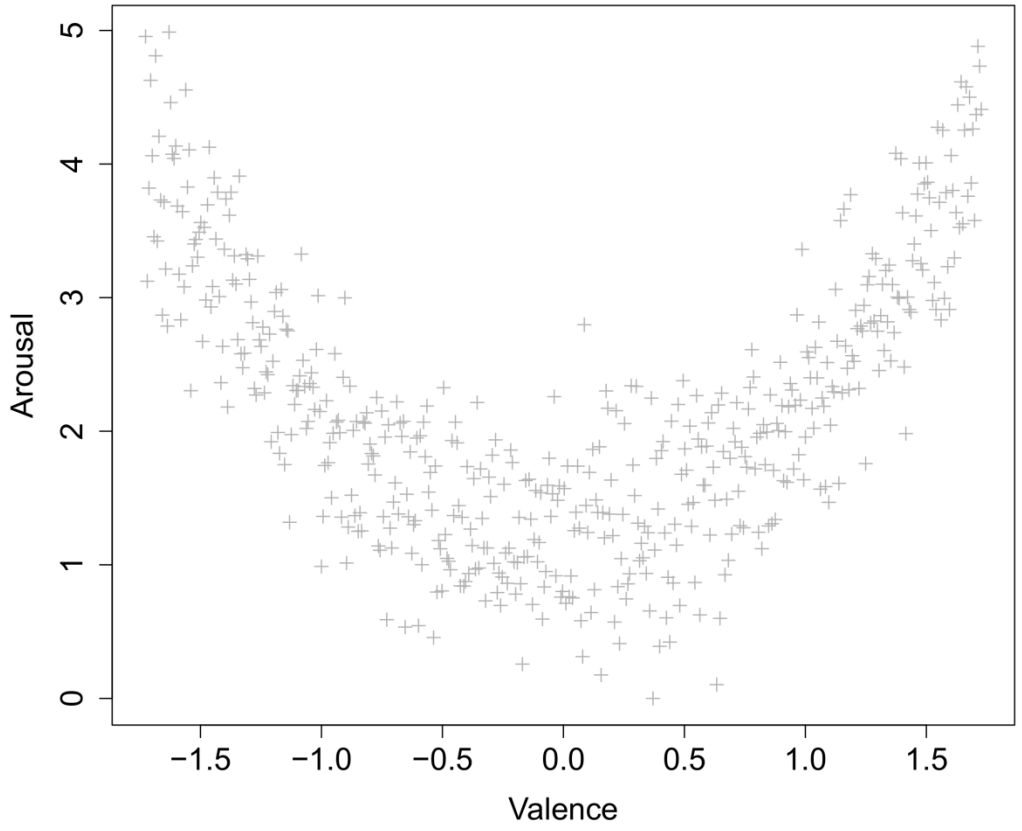
- Relation between the felt valence (positive – negative) of a stimulus and its felt arousal is thought to be quadratic (e.g., IAPS picture database):



(Lang, Bradley, Cuthbert, 1999)

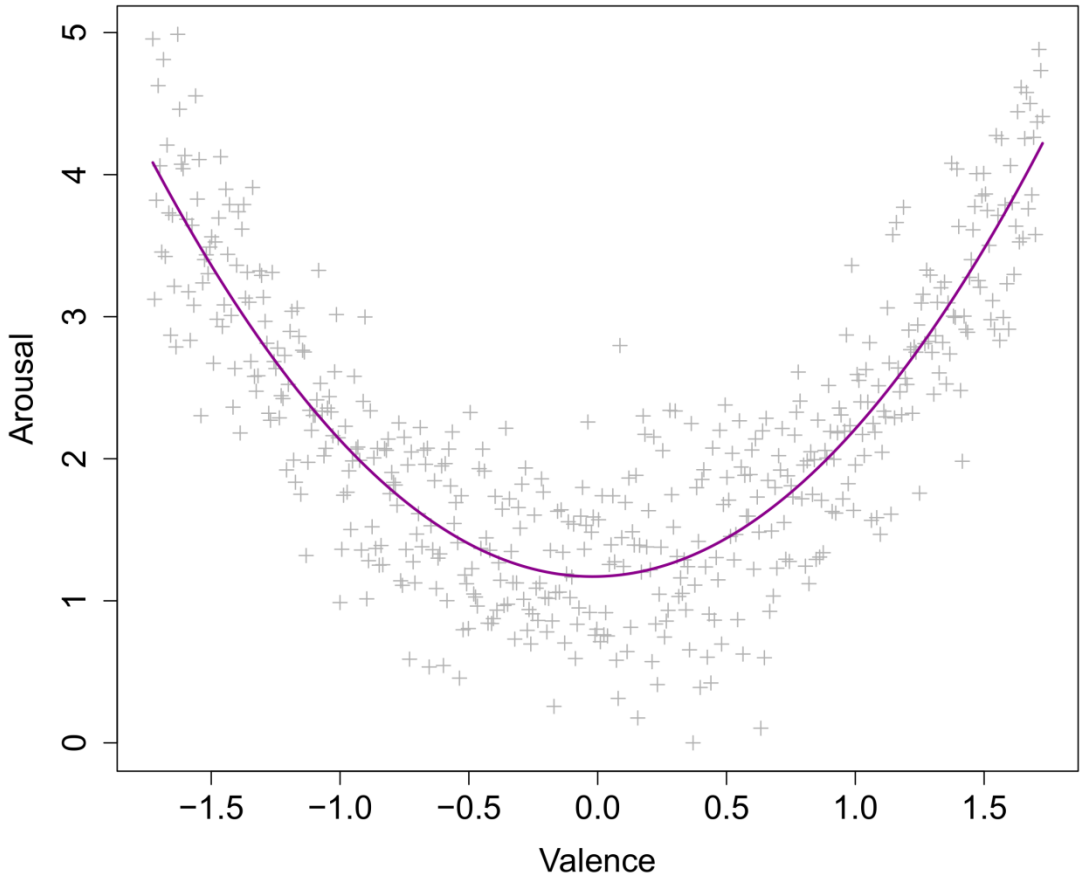
Dealing with curves – A problem

- Relation between the felt valence (positive – negative) of a stimulus and its felt arousal is thought to be quadratic (e.g., IAPS picture database):



(Lang, Bradley, Cuthbert, 1999)

Dealing with curves – A problem

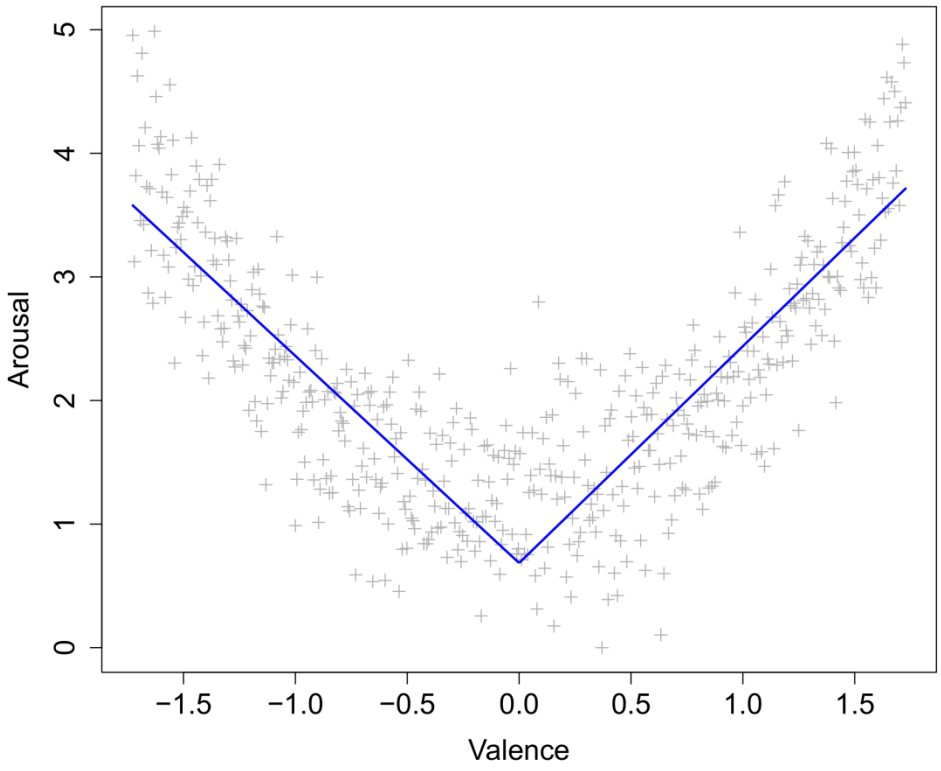


$$E(Y) = \beta_0 + \beta_1 X + \beta_2 X^2$$

What is the interpretation of β_2...?

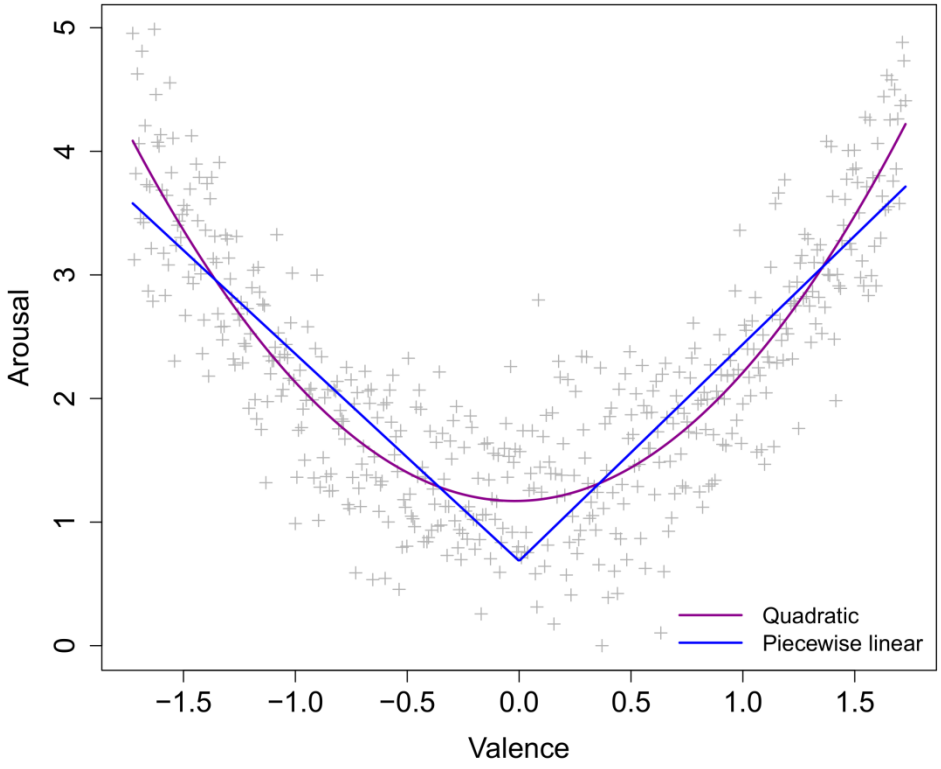
Dealing with curves – A solution?

- Unfortunately, parameters related to curves or even nonlinear parameters are difficult to interpret. One solution is to approximate a curve by a piecewise linear function:



Dealing with curves – A solution?

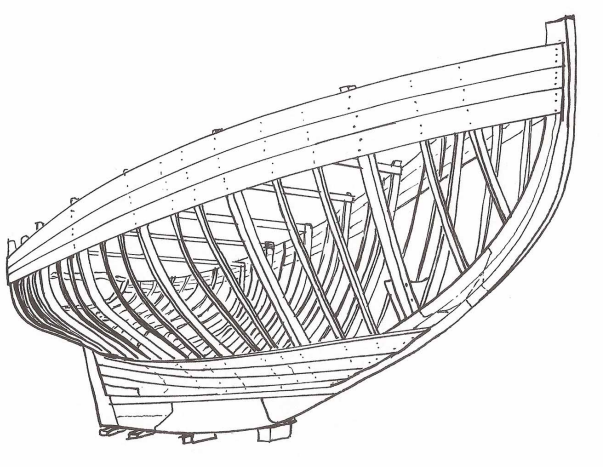
- As an approximation, this model is actually quite good. The interpretation is identical to that of linear slopes, except now we have a different slope before and after the change point.



Piecewise regression with splines

- Piecewise functions are called splines. In the previous example, we fitted a linear spline, due to the pieces being linear.

- Often the pieces of splines are not linear, but the linear type is especially attractive due to its simple interpretation (i.e., constant rate of change for each piece).



- Splines are joined together by knots, which are sometimes also called hinges. These are the points where the function can change direction.

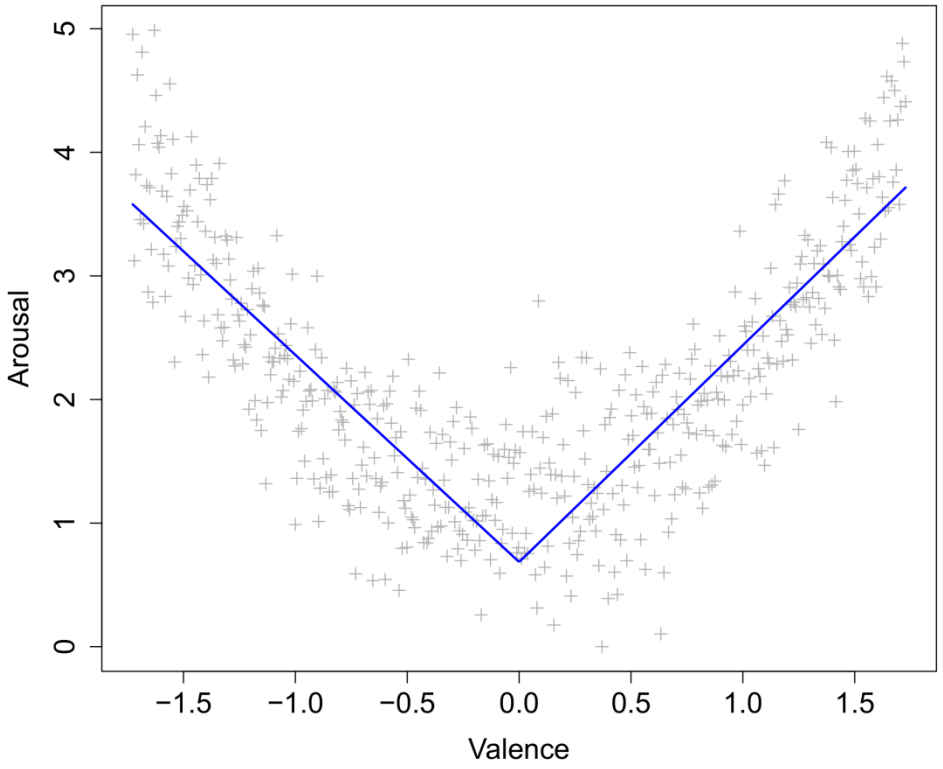
- In practice, it is easy to fit a simple spline model to data for a chosen knot point. We do this by adding hinge transformations of predictors to our data set.

Piecewise regression with splines

- The general notation for a hinge transformation is $(X-k)_+$, which denotes the positive part of X after the value of the knot k was subtracted.
- For the transformed variable, values smaller than the knot value will be set to zero. Values larger will be non-zero. Specifically:
  - $X < k \qquad X_{\text{hinge}} = 0$
  - $X > k \qquad X_{\text{hinge}} = X - k$
- Now we add this transformed variable as a predictor to a standard linear model…

X	$(X-0)_+$	Y
-2.0	0.0	5.1
-1.5	0.0	3.4
-1.0	0.0	2.3
-0.5	0.0	1.5
0.0	0.0	0.7
0.5	0.5	1.4
1.0	1.0	2.4
1.5	1.5	3.5
2.0	2.0	5.2

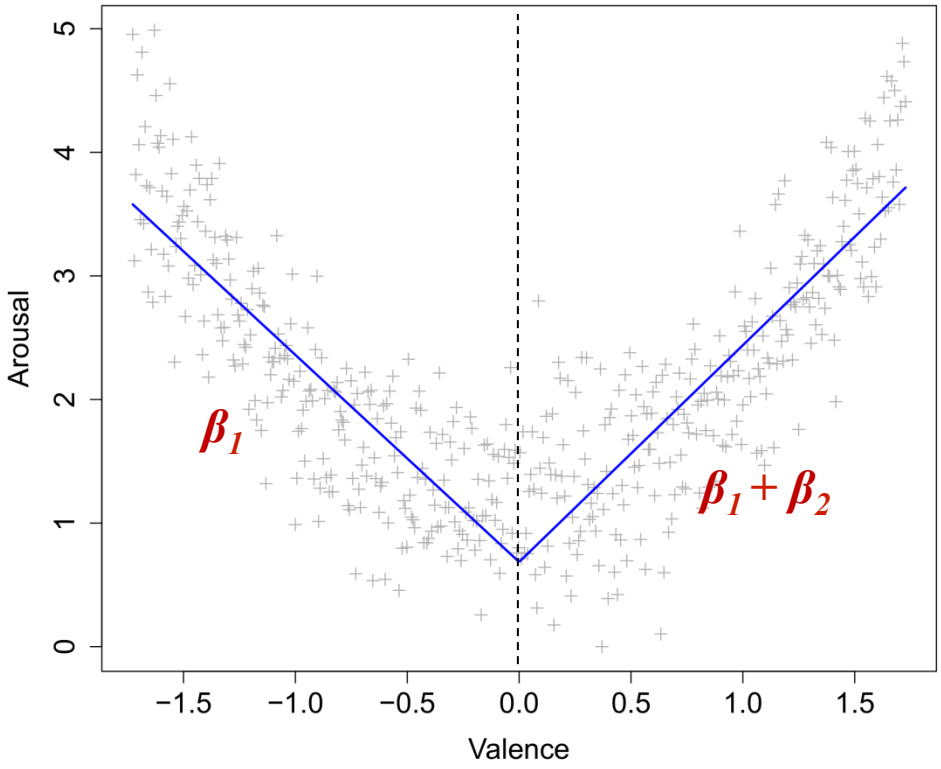
Piecewise regression with splines



$$\mathrm{E}(Y) = \beta_0 + \beta_1 X + \beta_2 (X - 0)_+$$

What is the interpretation of β_2...?

Piecewise regression with splines



$$E(Y) = \beta_0 + \beta_1 X + \beta_2 (X - 0)_+$$

β_2 reflects the amount of change in β_1 after the point X=0.

Piecewise regression with splines

- Significance tests on spline parameters thus reflect whether or not the current slope is significantly different from the preceding slope. In other words, whether there is significant *change* after a certain point.

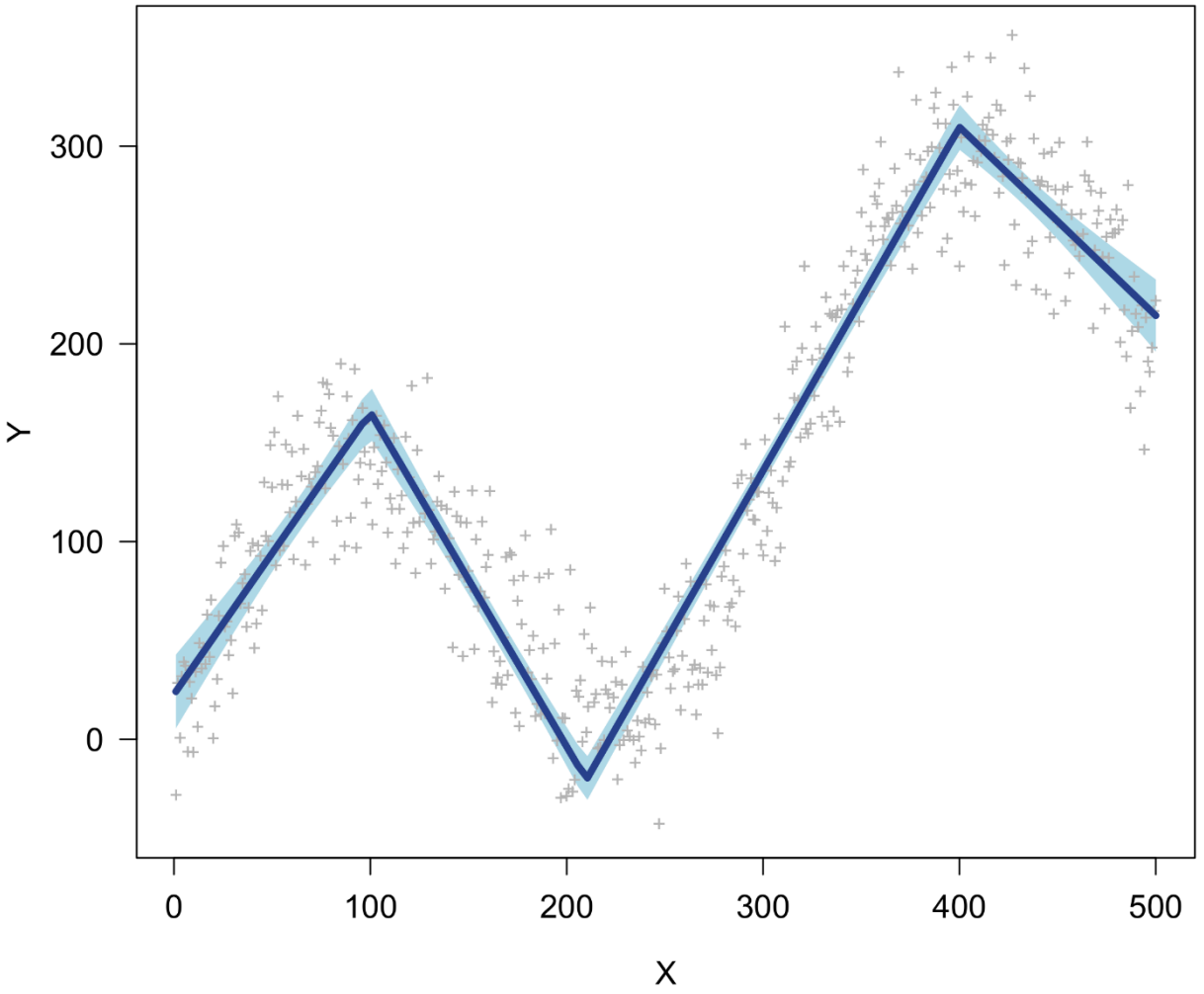
- In R, it is easy to manually create a hinge transform function, and use it to fit a linear spline for the curvilinear data:

```
h <- function(v,k=0) { ifelse(v-k<0,0,v-k) }
lspline <- lm( y~x+h(x,100)+h(210)+h(400) )
```

- R allows you to plug custom functions directly inside formulas. What does the model look like…?

```
visreg(lspline,xvar="x")
summary(lspline)
```

Piecewise regression with splines



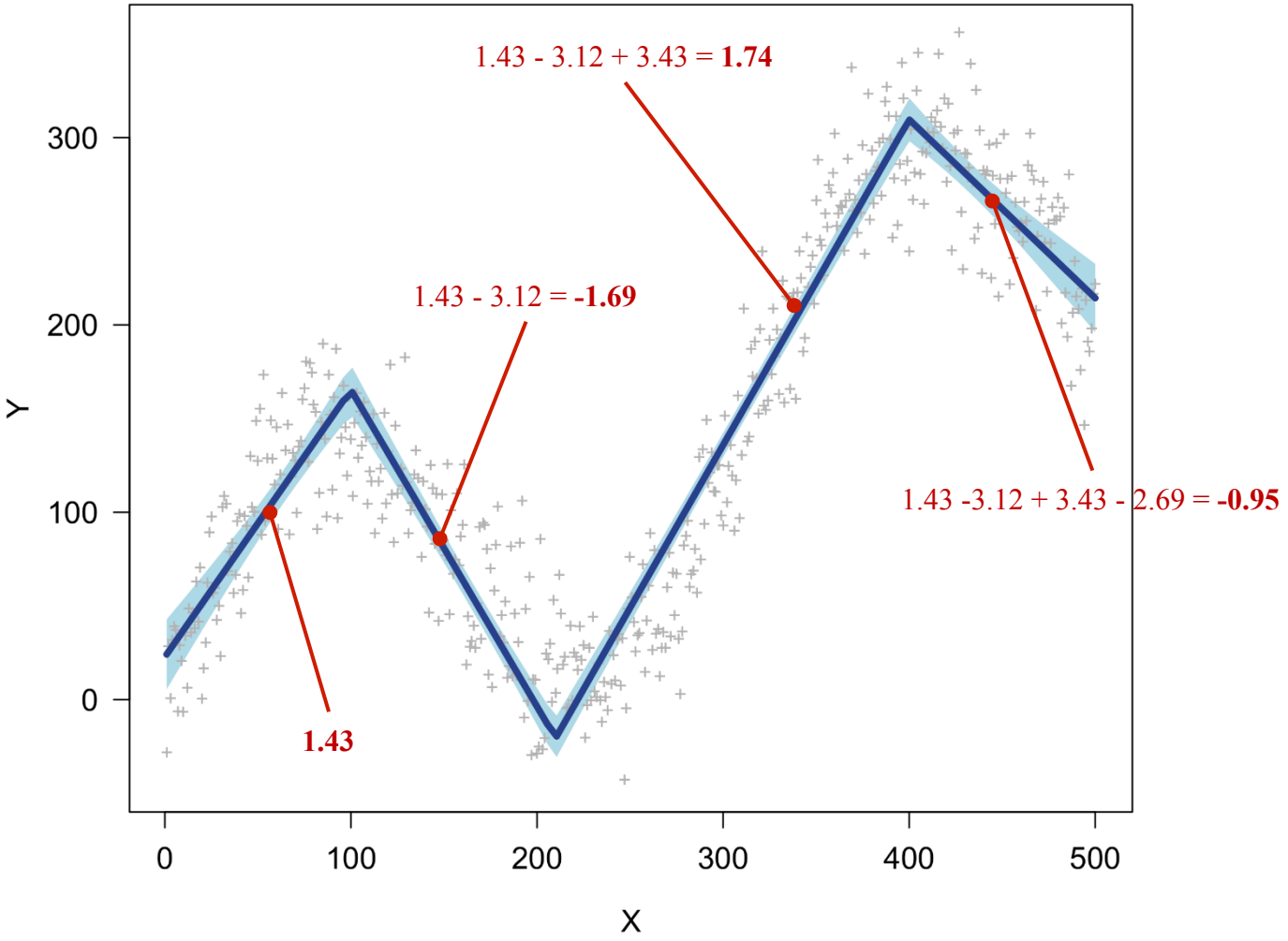
Piecewise regression with splines

- This time the parameters and their significance tests have a much more straightforward interpretation:

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 22.71886    5.69921   3.986 7.72e-05 ***
x            1.42673    0.08058  17.705  < 2e-16 ***
h(x, 100)   -3.11756    0.12094 -25.777  < 2e-16 ***
h(x, 210)    3.42884    0.07452  46.011  < 2e-16 ***
h(x, 400)   -2.69139    0.09370 -28.723  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 30.5 on 495 degrees of freedom
Multiple R-squared:  0.9069,    Adjusted R-squared:  0.9062
F-statistic:  1206 on 4 and 495 DF,  p-value: < 2.2e-16
```

Piecewise regression with splines



Piecewise regression with splines

- Note that the spline model has the same number of parameters as the polynomial fitted earlier (4), but the transformed X variables differ completely. A formal ANOVA comparison (with an F-test) is not possible between these two models, since they are not nested.

- However, in such a scenario, we could compare goodness-of-fit measures that do not require this nesting, such as adjusted R^2, or the information criteria AIC and BIC. All of these measures penalize models for redundant parameters. In R:

```
AIC(poly4) ; AIC(lspline)
> 4920.452
> 4843.655
```

- The linear spline model has an AIC that is lower by almost 80 points, which is a dramatic difference on a relative AIC scale, where differences of 2 points are typically already considered “meaningful”.

B-splines

- While the simple hinge transformations will return a highly interpretable model, the resulting variables might be vulnerable to instabilities, chiefly collinearity between them.

- For this reason, one might prefer a B-spline transformation, which is a kind of "normalized" spline transformation. In R*, this can be fitted as follows:

```
bspline <- lm(y~ bs(x,knots=c(100,210,400),degree=1) )
```

- This model will reduce the collinearity issue while otherwise having the exact same fit (e.g., R^2, AIC, visual fit).

- However, the parameters are much more complicated to interpret than the simple splines presented before…!

* requires `splines` package

6. When are splines useful?

When are splines useful?

- A need for splines can arise generally from the wish to model nonlinear associations. However, there are some specific cases where splines are particularly useful:

    1. Bipolar scales/factors

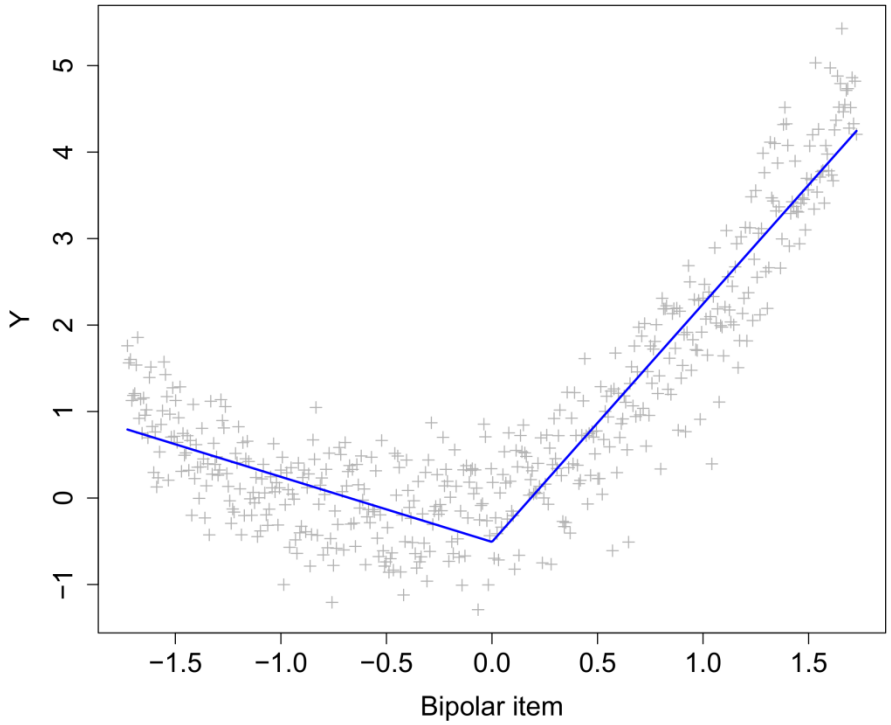
    2. Threshold effects

    3. Longitudinal change

Bipolar scales/factors

- Questionnaire items in psychology often consist of bipolar scales (negative–positive; disagree–agree). However, the negative part of these scales may have a different relationship with a response variable than the positive part

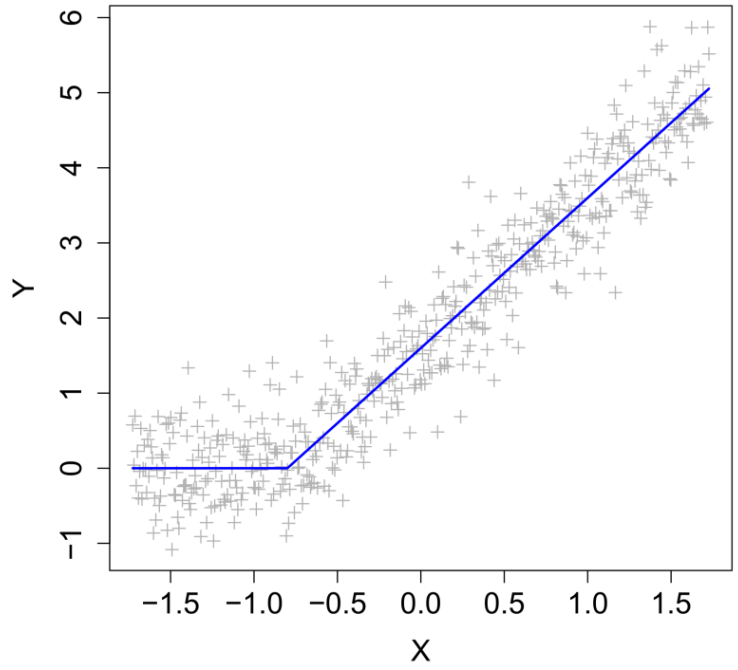
- Splines allow these parts to be modelled separately

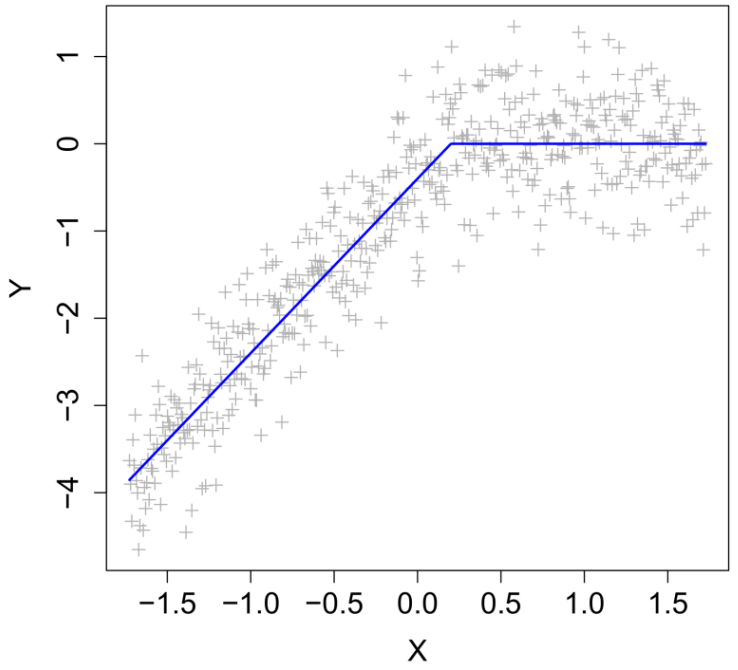


- Bipolarity is also typically observed in latent variables such as principal components in PCA and factors in factor analysis!

Threshold effects

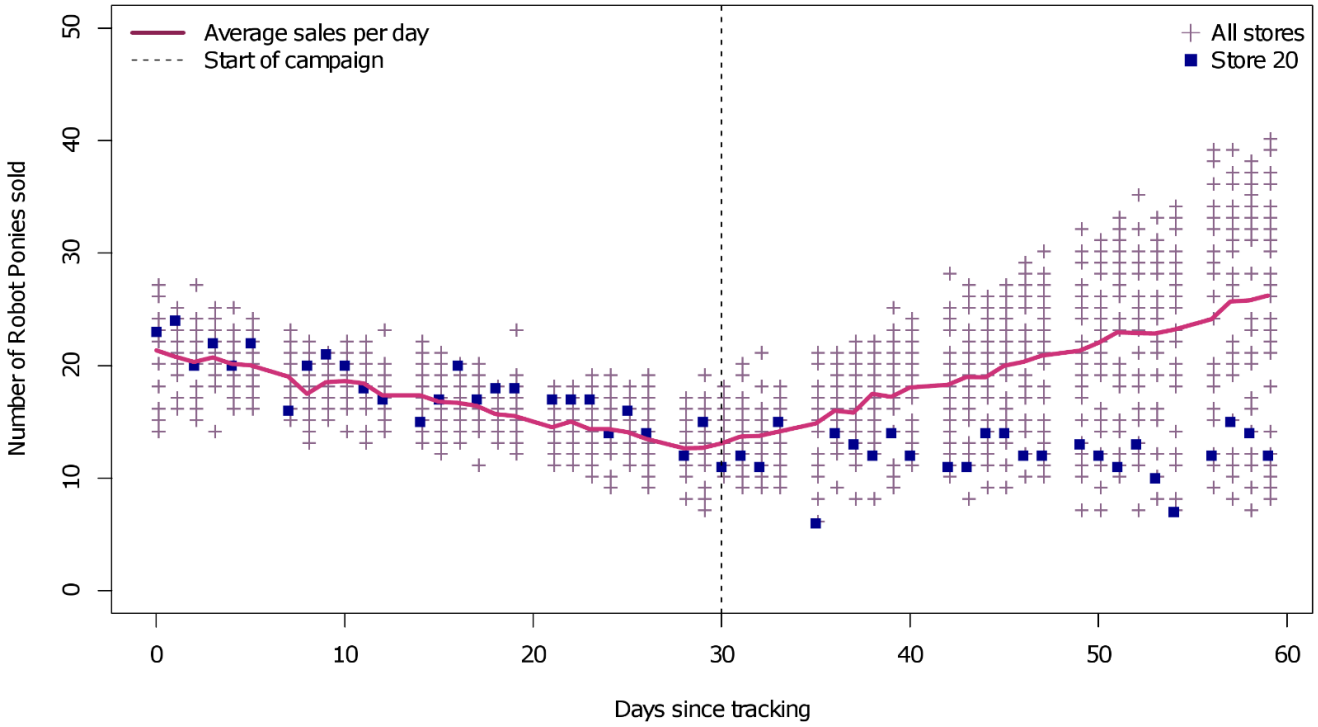
- Sometimes predictors do not influence a response variable until a certain value is reached. Sometimes predictors stop influencing a response variable after they reach a certain value:





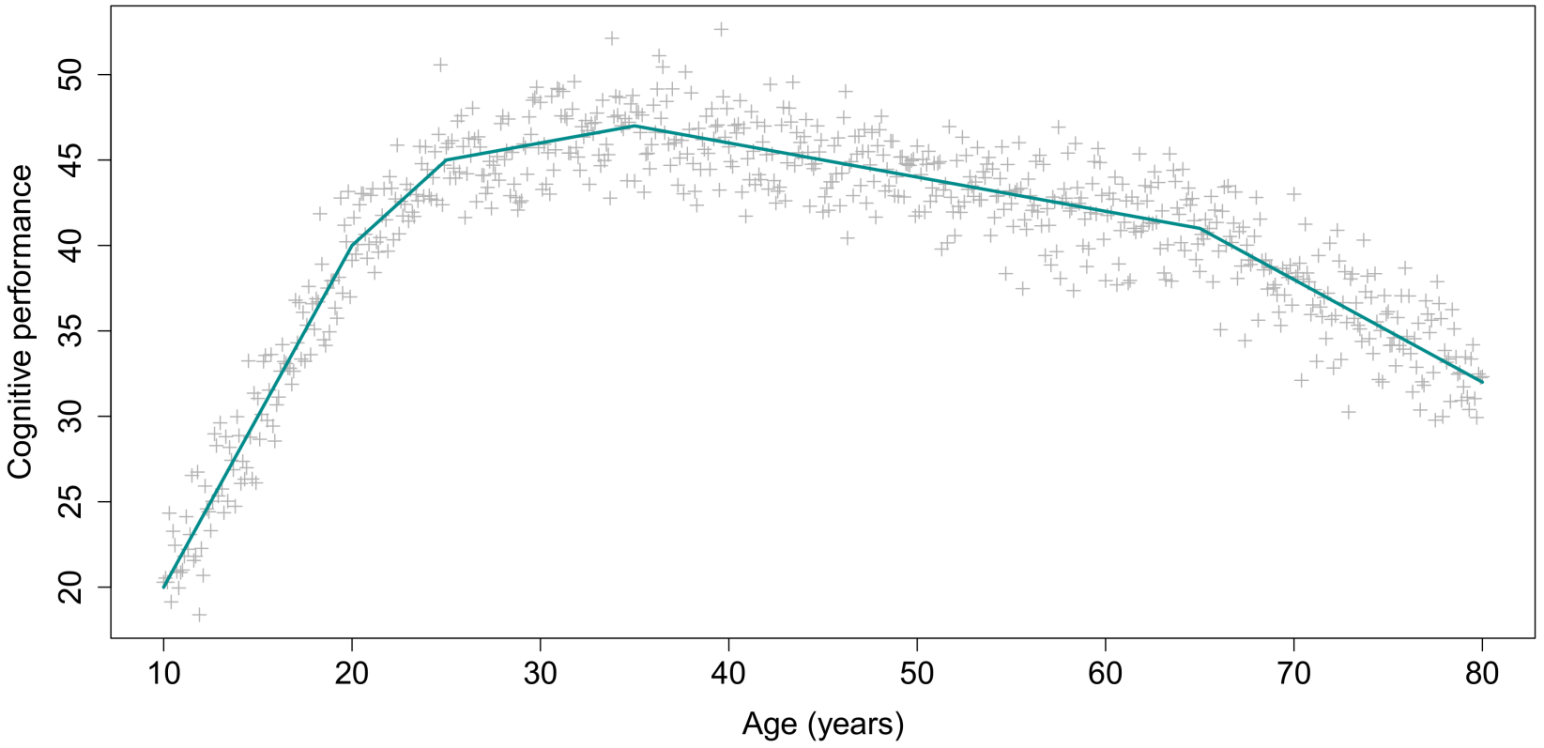
Longitudinal change

- Change over time is rarely characterized by a single linear slope. Typically, longitudinal data are characterized by alternating phases of stability and change. Changes can be a consequence of events or explicit interventions:



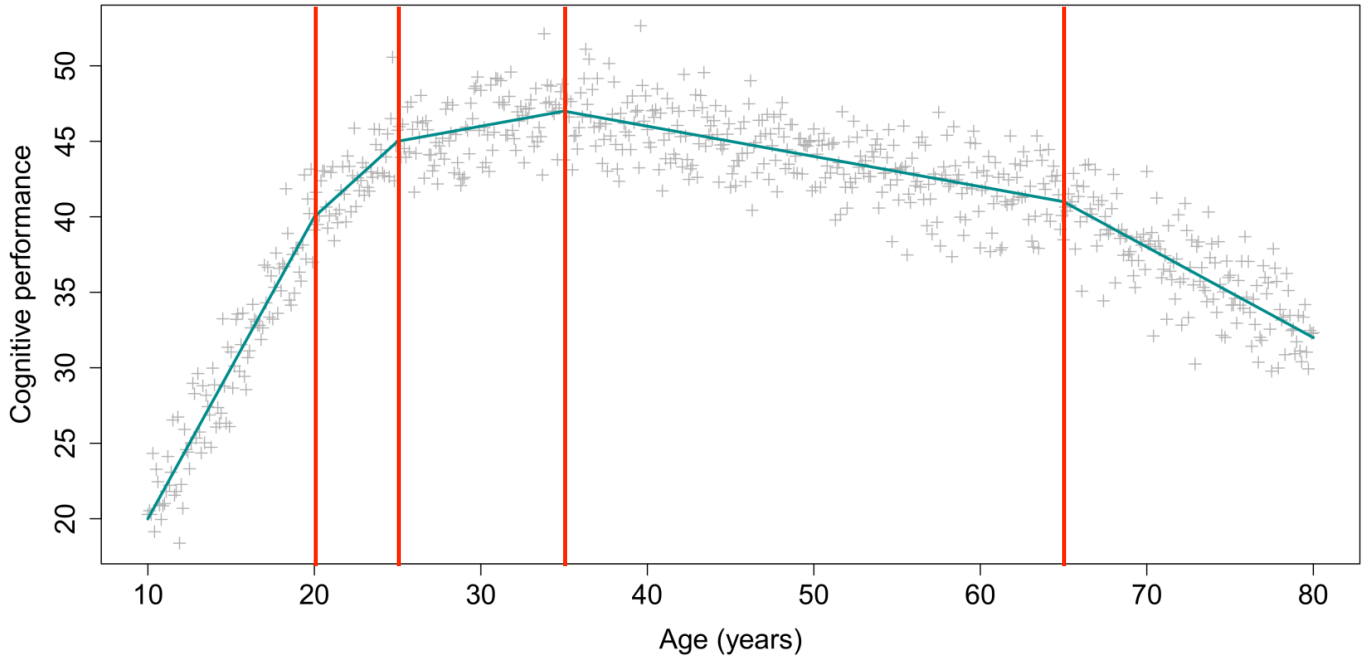
Longitudinal change

- Alternatively, longitudinal data could have been collected without the presence of interventions, or *time* reflects a cross-sectional pattern of change:



Data format

- Splines allow a variable to be simultaneously categorical and continuous. Each spline represents a category, but within that category there can be a continuous relationship between the predictor and the response. The knots ensure that the resulting function has no discontinuities.



The problem of selecting knot points…

- So far in this discussion we have ignored a rather massive problem. How exactly do we select *appropriate* knot points for our predictor variables? How do we even know for sure if our predictor variables need splines?

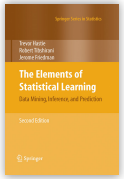
- For the previous examples the knot values could be selected a priori, based on theoretical or psychometric justifications:

  - The mid-point of a bipolar variable
  - A point of intervention in longitudinal data
  - Pre-defined age categories (e.g., adolescent, young adult, old adult)

- Often, however, it will not be clear where the knots should be placed. In this case, we should consider a data-driven selection of knots. This is the primary purpose of multivariate adaptive regression splines (MARS; Friedman, 1991)…

7. Multivariate Adaptive Regression Splines

(Tibshirani et al., 2009; Chapter 9)



Multivariate adaptive regression splines

- MARS models a dependent variable (Y), given one or more independent variables (X's), just as linear regression. In addition, it automatically adds meaningful main effects, interactions, and hinge transformations to the model.

- This MARS achieves with a stepwise selection procedure that selects:

    1. Relevant predictor variables
    2. The optimal number of knot points per predictor
    3. The optimal values for the knot points pre predictor
    4. Interactions between predictors up to a specified degree

- These elements are added or removed only when they reduce the model's generalized cross-validation error (GCVE), which is a penalized measure of model error similar to AIC and BIC.

MARS in R

- For the curvilinear data, fitting a MARS model is straightforward:

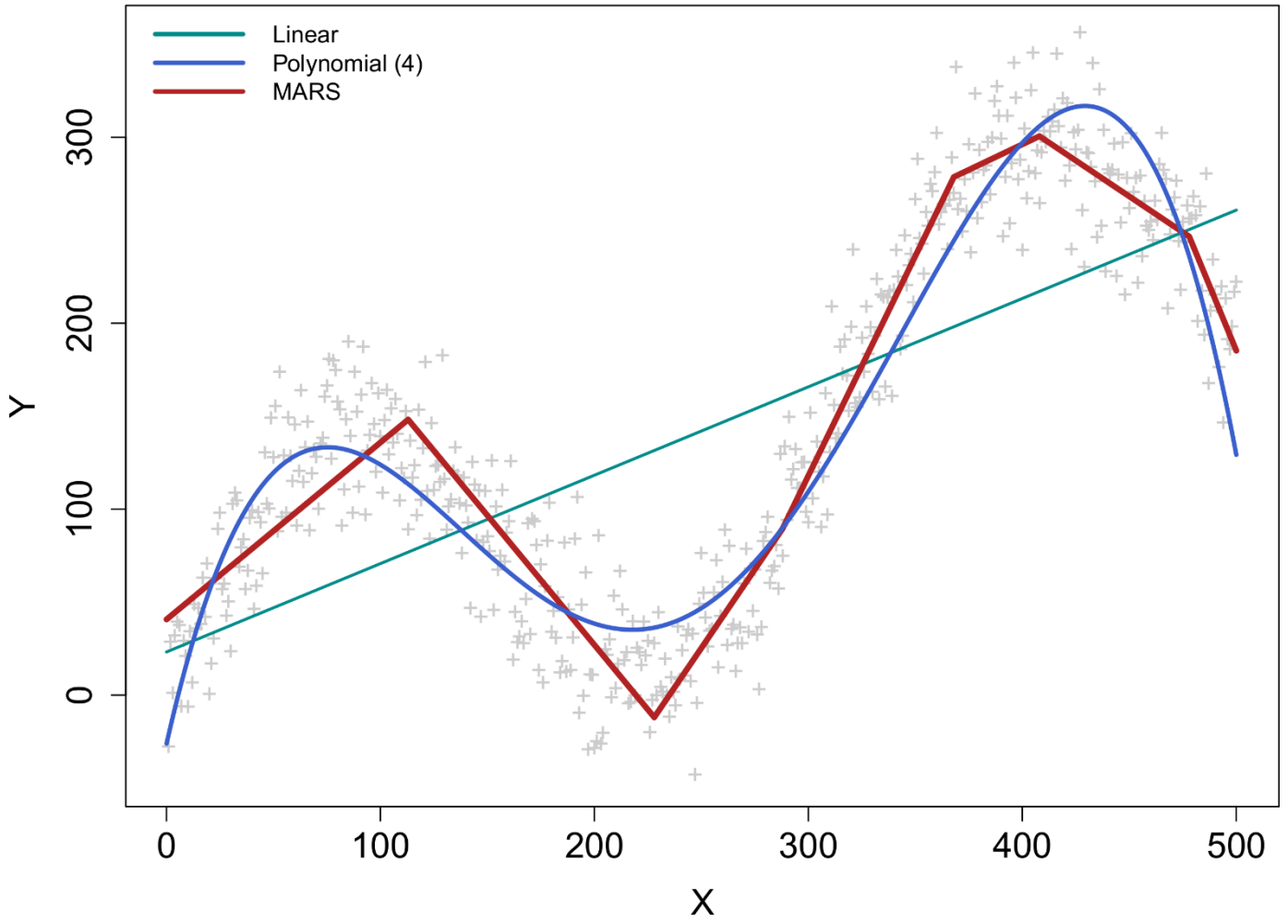
```
library(earth)
mars <- earth(y~x)
```

- Unfortunately, `visreg` is incompatible with the `earth` package, so we might switch to manual prediction plotting:

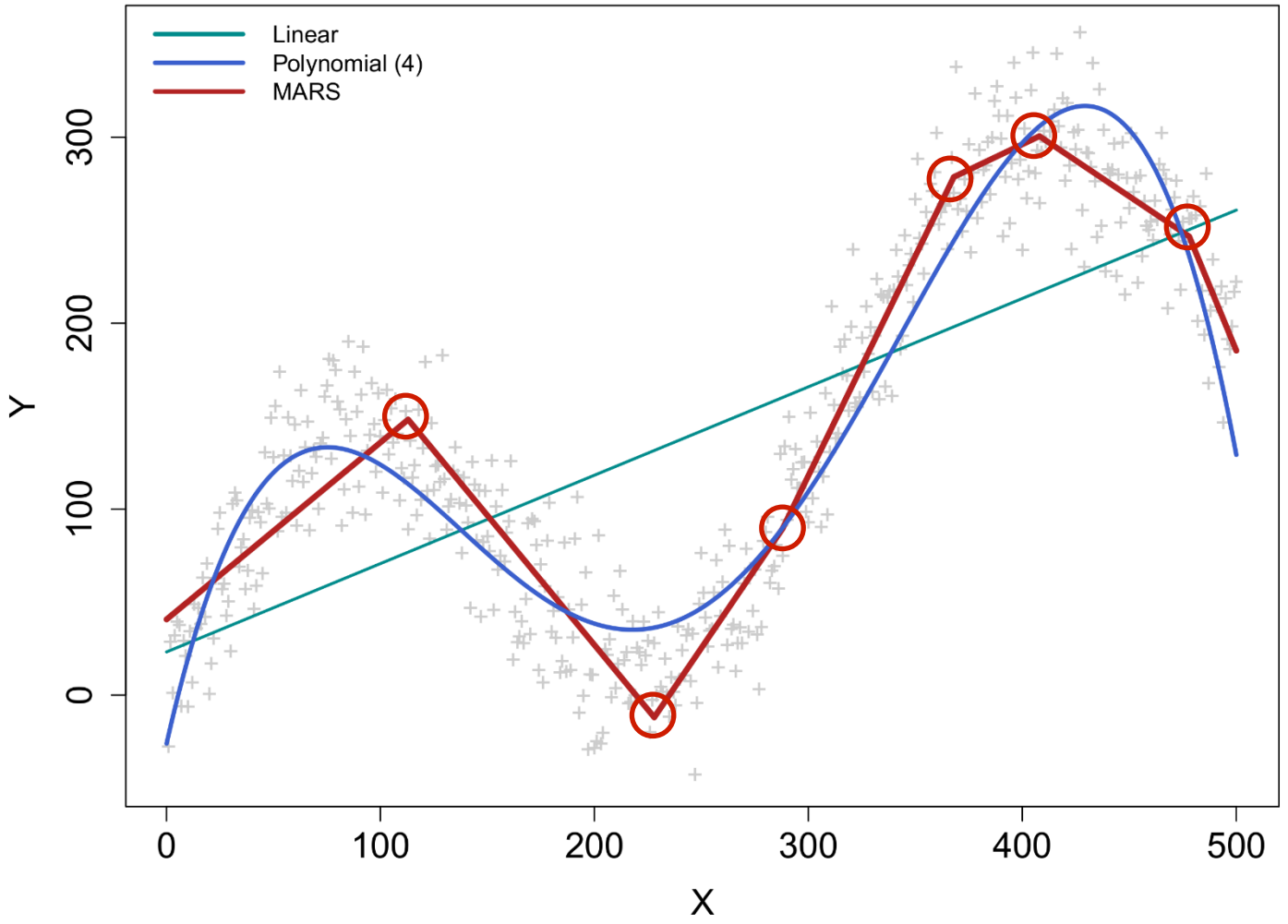
```
xval <- 1:5000/10
predmars <- predict(mars,newdata=data.frame(x=xval))
plot(x,y,pch="+",col="grey80",xlab="X",ylab="Y")
lines(xval,predmars,col="firebrick",lwd=4)
```

- An alternative is to use `earth`'s compatible `plotmo` package, which offers some of the same functionalities as `visreg`.

MARS in R



MARS in R



MARS in R

- The summary prints the exact values of the slopes in the different value ranges of X, as well as some other highly useful model information:

```
             coefficients
(Intercept)   257.656056
h(x-113)       -2.344439
h(228-x)       -0.951901
h(x-228)        4.036941
h(x-288)        0.670482
h(x-368)       -1.814359
h(x-408)       -1.319833
h(x-478)       -2.017451

Selected 8 of 8 terms, and 1 of 1 predictors
Termination condition: RSq changed by less than 0.001 at 8 terms
Importance: x
Number of terms at each degree of interaction: 1 7 (additive model)
GCV 879.8306    RSS 413916.3    GRSq 0.911449    RSq 0.9163481
```

A more interesting data set…

- The curve data is only interesting for the splines aspect of MARS. The model cannot harness its stepwise predictor and interaction selection. Let's look at the `etitanic` data set that's included in the `earth` package:

#	survived	pclass	sex	age	sibsp	parch	noise norm	noise unif	noise binom
1	yes	1st	female	29.00	0	0	-1.2933	-0.5936	0
2	yes	1st	male	0.92	1	2	-1.8854	0.8110	1
3	no	1st	female	2.00	1	2	0.1692	-0.2348	0
4	no	1st	male	30.00	1	2	-0.2159	-1.4384	0
…	…	…	…	…	…	…	…	…	…
1046	no	3rd	male	29.00	0	0	0.9123	-0.3864	1
Index	DV	IVs					Noise IVs		

Survival on the Titanic

- We want to predict/model the probability of survival (0=no, 1=yes) based on the passenger information. We want to select the best predictors for this, their interactions (if any), and any spline effects (if any).

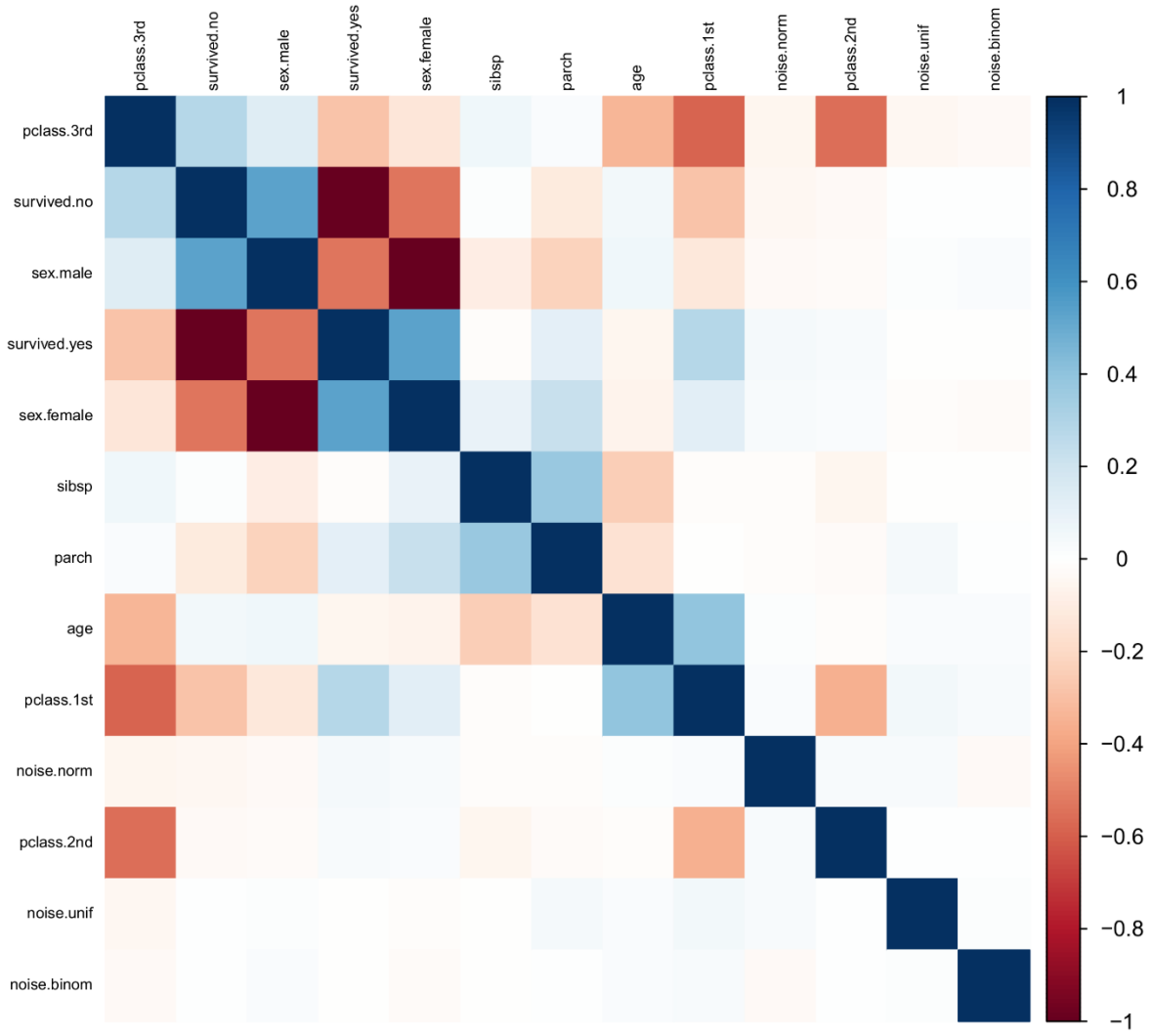
- Since we have a binary dependent variable, the underlying model will be a logistic regression. However, MARS has no problem handling this scenario, and will apply its predictor and knot selection as before.

- To make the Titanic data slightly more challenging for the model, I added three random noise variables (Gaussian, uniform, and binomial noise).* Ideally, MARS's stepwise selection procedure should exclude these variables from the final model.

- Adding such variables to a dataset can be useful way of detecting problems with overfitting!

* Full R code for data preparation in appendix

Survival on the Titanic



requires `corrplot` package

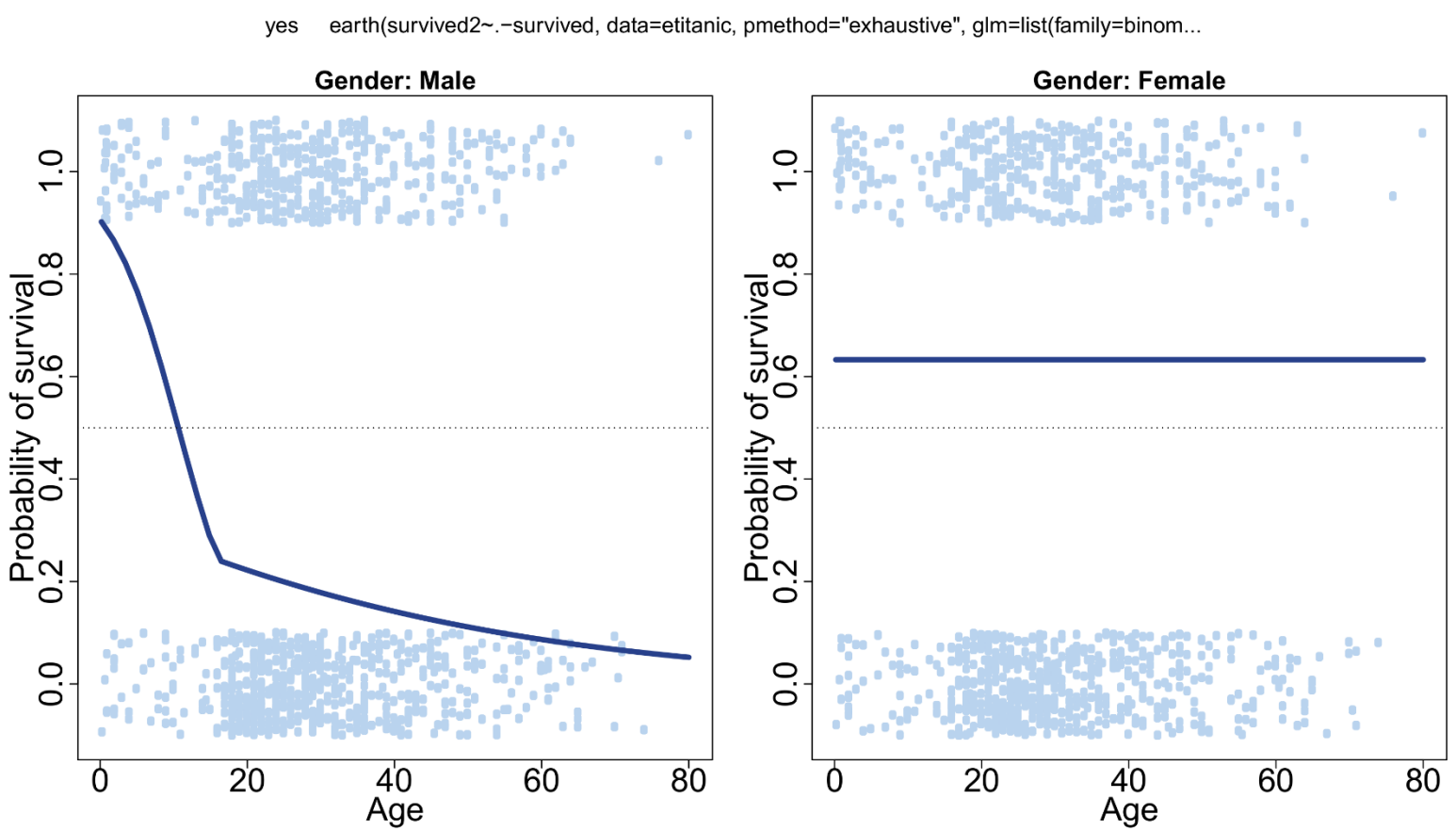
Survival on the Titanic

```
mars <- earth(survived~.,data=etitanic,degree=2,pmethod="exhaustive",
  ,glm=list(family=binomial))
summary(mars)
```

```
                                GLM coefficients
(Intercept)                            2.7189042
pclass3rd                             -4.3236171
sexmale                               -2.6807708
pclass2nd * sexmale                   -1.7478078
pclass3rd * sexmale                    0.9930032
pclass3rd * h(3-sibsp)                 0.7209374
pclass3rd * h(0.0180345-noise.norm)   -0.4315225
sexmale * h(age-16)                   -0.0274905
sexmale * h(16-age)                    0.2124101
h(1-parch) * h(noise.unif-1.55032)    -4.5559464

Earth selected 10 of 17 terms, and 8 of 9 predictors
Termination condition: Reached nk 21
Importance: sexmale, pclass3rd, pclass2nd, age, sibsp, parch,
noise.unif, noise.norm, noise.binom-unused
Number of terms at each degree of interaction: 1 2 7
Earth GCV 0.140843    RSS 140.7765    GRSq 0.4180988    RSq 0.4428869
```

Age × Gender interaction



* Full R code in appendix

Predictor relevance in MARS

- MARS returns multiple indexes of predictor importance in its final model. The most important is simply which predictors were actually used.

- In addition, MARS ranks their importance by their obtained decrease in GCVE, relative to the most important predictor (which is set to 100). This ranking can be accessed via the `evimp` function:

```
evimp(mars)
>             nsubsets   gcv    rss
> sexmale            9 100.0  100.0
> pclass3rd          8  56.2   58.9
> pclass2nd          6  36.9   40.5
> age                6  36.9   40.5
> sibsp              5  26.0   30.6
> parch              2   4.8   12.0
> noise.unif         2   4.8   12.0
> noise.norm         1   1.5    8.0
```

Predictor relevance in MARS

- For continuous predictors, another index of predictor relevance can be the number of knots that were selected, with more knots reflecting more nonlinearity in that predictor's relationship with the dependent.

- Here, only age really qualifies for inspection. For males, it appears there is a drastic increase in survival probability for passengers younger than 16. No further knot points were relevant.

- Disturbingly, two of the noise variables also appear in the final model! While they are ranked as least important by the `evimp` ranking, their appearance is still highly undesirable.

- Perhaps we can tweak the basic model further to prevent this…?

Control parameters

- The earth implementation of MARS has a rather large number of different control options and tuning parameters (see its help pages for explanations). Some of the important ones include:

Argument	Details
`degree`	The order of interactions among predictor variables that you allowed. Typically values between 2 and 4 make the most sense
`pmethod`	The procedure for stepwise selection, which can be forward, backward, exhaustive, or fully cross-validated
`nfold`	The number of cross-validation folds to consider when pmethod="cv"
`penalty`	A penalty value for adding knots to the model. Larger values will reduce the number of knots MARS includes in the final model. Simulation studies suggest values in the range of about 2 to 4. 0 means no splines.

- In addition, the user has custom control over which variables are allowed to (a) be dropped, (b) interact, and (c) have spline effects.

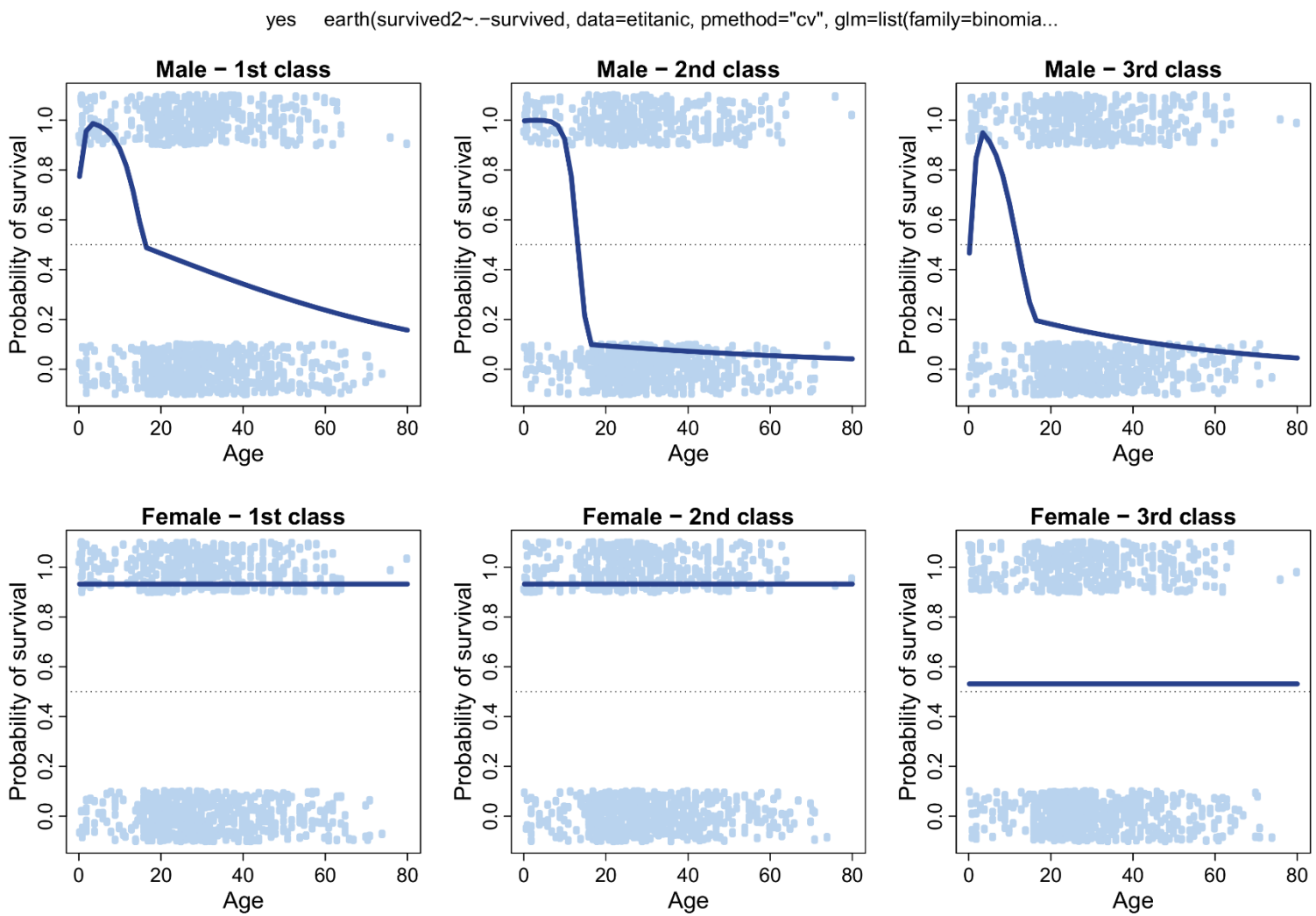
Cross-validated selection

```
set.seed(1846)
mars <- earth(survived~.,data=etitanic,degree=3,pmethod="cv",
  glm=list(family=binomial),nfold=5)

(Intercept)                            2.6210388
pclass3rd                             -3.6762798
sexmale                               16.5330822
pclass2nd * sexmale                   -2.1665179
pclass3rd * sexmale                    1.1304006
pclass3rd * h(sibsp-3)                -0.3812213
pclass3rd * h(3-sibsp)                 0.6411103
sexmale * h(age-16)                    1.4505704
sexmale * h(16-age)                   -1.1718619
sexmale * h(age-3)                    -1.4762406
pclass2nd * sexmale * h(age-16)        0.0111782
pclass2nd * sexmale * h(16-age)        0.4326142
pclass3rd * h(3-sibsp) * h(parch-1)   -0.2300371
pclass3rd * h(3-sibsp) * h(1-parch)   -0.2478007
sexmale * h(16-age) * h(sibsp-1)      -0.0858618
sexmale * h(16-age) * h(1-sibsp)       0.0398533

Earth selected 16 of 16 terms, and 6 of 9 predictors using pmethod="cv"
Termination condition: Reached nk 21
Importance: sexmale, pclass3rd, pclass2nd, age, sibsp, parch, noise.norm-unused,
noise.unif-unused, noise.binom-unused
Number of terms at each degree of interaction: 1 2 7 6
Earth GRSq 0.4143498  RSq 0.4556279  mean.oof.RSq 0.4211055 (sd 0.0423)
```

Age × Gender × Class interaction



MARS cautions

- MARS does not output significance tests for its selected effects. Whether or not an effect (a knot, an interaction) is meaningful is determined by the stepwise selection with GCVE.

- If there is an interest in testing a knot effect for significance, it is better to add the spline effect manually to your data and run the analysis with a conventional program/function (even in SPSS or Statistica you can do this).*

- In general, MARS will perform better for larger data sets, and be more stable for uncorrelated independent variables. Stepwise selection in the presence of severe confounding (and especially multicollinearity) can be problematic!

- MARS should not be used to analyze data from experimental designs. It is best suited to observational data and for data exploration, for example as a follow-up to a main analysis.

*Or plug the model matrix directly into a regular regression function in R

More earth…

- If you consider a MARS/spline approach for your own data, I highly recommend to look up the `earth` package's help pages. Its documentation and vignettes are some of the most informative of any package available in R!

- See especially, "Notes on the earth package": http://www.milbo.org/doc/earth-notes.pdf

- The notes PDF answers every conceivable question about the algorithm, parameter interpretation, plotting, goodness-of-fit, etc.

- In addition, earth offers extensions for other types of response distributions (multinomial, poisson) and even multivariate data (i.e., more than one response variable simultaneously). Finally, there is an option model the variance of your data, rather than its mean (e.g., when there is heteroscedasticity).

Why should I use splines?

- Linear spline models can be justified easily due to their high interpretability when compared to curves. Moreover, there may be sound theoretical reasons to allow knot points in the data (e.g., bipolar scales, longitudinal events).

- If the values of the knot points can be set based on a-priori grounds, this is generally recommended.

- If the required number of knot points and splines is unknown, MARS is a powerful machine learner that will automate this selection. The same model can be applied to search for promising interactions. An exhaustive search for interactions with inferential tests (*p*-values) would be practically unfeasible.

- MARS has not been widely adopted in many sciences, to date, although its core principles should be either familiar to researchers (stepwise regression) or easily relatable (splines).

MARS evaluation

- Advantages:
  - + Models retain their linear interpretability
  - + Automated selection of splines and interaction effects
  - + Multiple rankings of predictor importance (not a black box!)
  - + Extremely well-documented earth package in R
- Disadvantages
  - – Unstable in data sets with correlated/collinear predictors
  - – Tends to generate overly complex models in large data sets
  - – Tends to generate overly simplistic models with cross-validated pruning
  - – Sensitive to the presence of many irrevelant/noise predictors
  - – Not generally applicable to experimental data

8. Conclusions

Machine learning and today's goals

- I started this presentation with a general introduction on machine learning, and how this is connected to linear regression.

- Following this, I showed how linear regression can easily be extended to afford flexibilities typically found in machine learning (nonlinear patterns).

- Piecewise linear regression, especially its implementation in MARS, is decidedly *not* a black box model! It was designed with the explicit purpose of being interpretable. This illustrates that not all ML need be incomprehensible.

- The next time you read about a media-hyped application of ML, you should wonder if the developers were not simply using MARS…

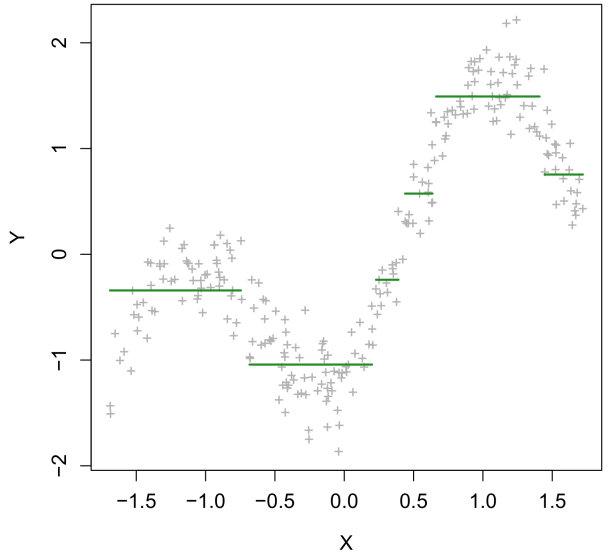
Splines and related models

- Linear splines are one convenient approach to extending the standard linear regression, but there are many others from the field of machine learning.

- MARS has similarities not only to stepwise linear regression, but also to decision trees (see my previous R Lunch). Whereas MARS fits a piecewise linear model, trees fit a piecewise constant model.

- A fully generalized version of MARS are Generalized Additive Models (GAM), which allow further flexibility in the choice of response distribution, and the transformation applied to the predictors.

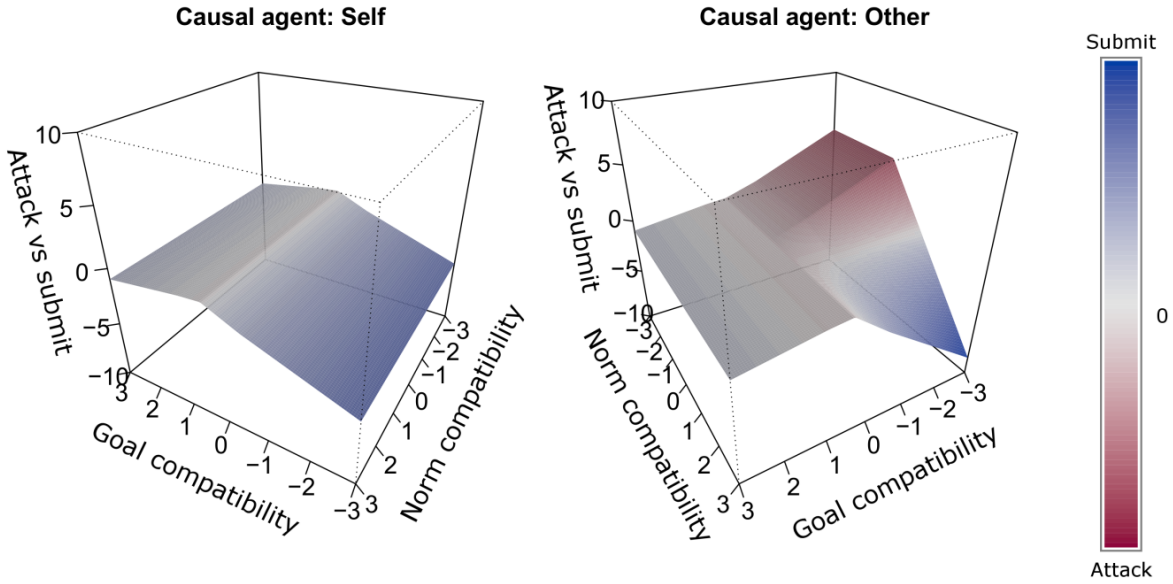
- Then there is so much more machine learning…



Thank you for your attention! Questions…?

Example – Action tendencies in emotion

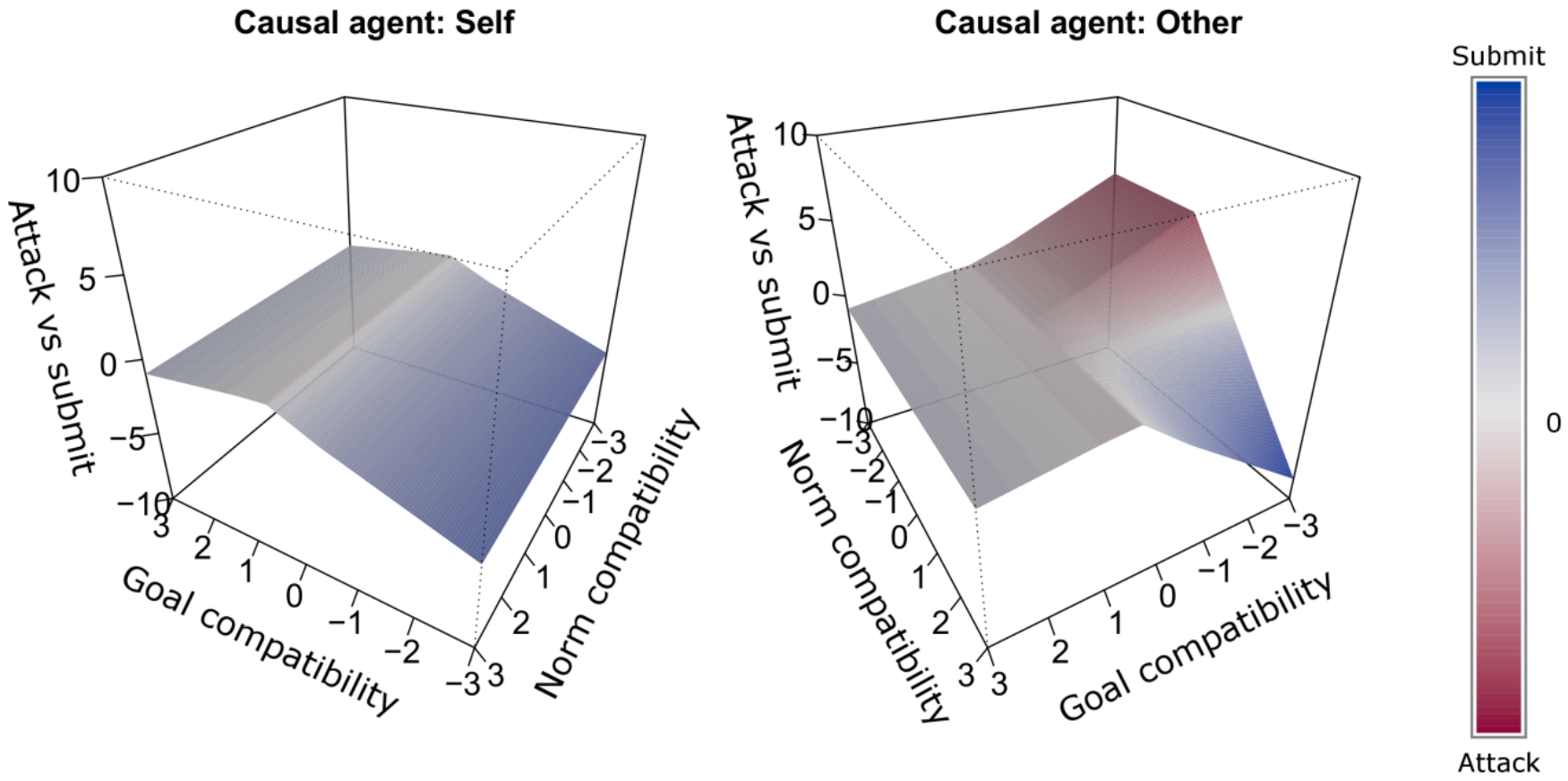
- We submitted a paper with *Emotion* that used multivariate adaptive regression splines (MARS) for modelling emotion data.



- The usage of these features could be motivated within appraisal theories of emotion which have predicted these types of nonlinearity.

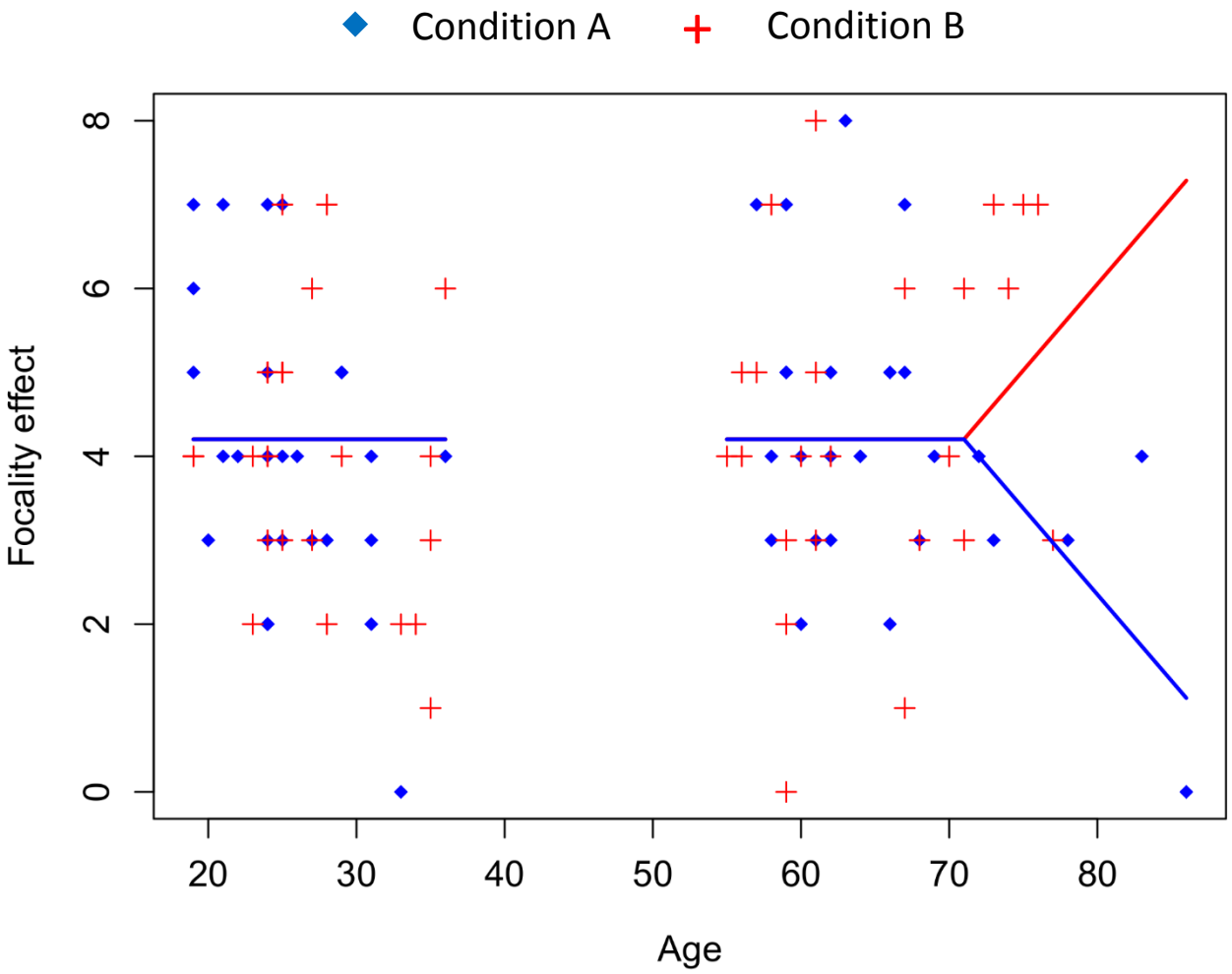
Meuleman et al. (2019)

Example – Action tendencies in emotion



Meuleman et al. (2019)

Example – Cognitive aging



Zuber, S., Ihle, A., Blum, A., Desrichard, O., & Kliegel, M. (2017)

R code for examples

```
library(earth)
library(plotmo)
library(visreg)
library(splines)
h <- function(v,k=0) { ifelse(v-k<0,0,v-k) }

## IAPS DATABASE EXAMPLE
set.seed(2333)
x <- scale(1:500)
y <- x^2+rnorm(500,0,0.5)
miny <- min(y)
y <- y-miny
xh <- h(x)
yq <- x+x^2+rnorm(500,0,0.5)
minyq <- min(yq)
yt1 <- 2*h(x,-0.8)+rnorm(500,0,0.5)
yt2 <- rev(-2*h(x,-0.2)+rnorm(500,0,0.5))

par(mar=c(5,5,1,1),cex.lab=1.5,cex.axis=1.5)
plot(x,y,pch=3,col="grey70",xlab="Valence",ylab="Arousal")
lines(x,x^2-miny,col="blue",lwd=2)
lines(x,fitted(lm(y~x+I(x^2))),col="darkmagenta",lwd=2)
lines(x,fitted(lm(y~x+xh)),col="blue",lwd=2)
legend("bottomright",legend=c("Quadratic","Piecewise
linear"),lwd=2,seg.len=3,bty="n",col=c("darkmagenta","blue"),cex=1.2)
```

R code for examples

```
## CURVILINEAR DATA PROBLEM
set.seed(668)
x <- 1:500
true <- 100*sin(0.02*x) + 0.5*x
y <- true + rnorm(500,0,25)
noise <- rnorm(500,0,10)
par(mar=c(5,5,1,1),cex.lab=1.2,cex.axis=1.2)
plot(x,y,pch="+",col="grey70",xlab="X",ylab="Y",cex=1.2)

linear <- lm(y~x)
poly4 <- lm(y~poly(x,degree=4))
lspline <- lm(y~x + h(x,100) + h(x,210) + h(x,400))

summary(linear)
summary(poly4)
summary(lspline)
AIC(poly4) ; AIC(lspline)

mars <- earth(y~x)

par(mfrow=c(1,2))
visreg(linear,xvar="x",line.par=list(col="royalblue4",lwd=4),points.par=list(cex=0.8,pch="+",col
="grey70"),fill.par=list(col="lightblue"),xlab="X",ylab="Y",alpha=0.001)
visreg(poly4,xvar="x",line.par=list(col="royalblue4",lwd=4),points.par=list(cex=0.8,pch="+",col=
"grey70"),fill.par=list(col="lightblue"),xlab="X",ylab="Y",alpha=0.001)
visreg(lspline,xvar="x",line.par=list(col="royalblue4",lwd=4),points.par=list(cex=0.8,pch="+",co
l="grey70"),fill.par=list(col="lightblue"),xlab="X",ylab="Y",alpha=0.001)
```

R code for examples

```
linear <- lm(y~x)
poly4 <- lm(y~poly(x,degree=4))
lspline <- lm(y~x + h(x,100) + h(x,210) + h(x,400))

mars <- earth(y~x)
summary(mars)

xval <- 1:5000/10
predlin <- predict(lin,newdata=data.frame(x=xval))
predpoly <- predict(poly,newdata=data.frame(x=xval))
predmars <- predict(mars,newdata=data.frame(x=xval))

par(mfrow=c(1,1),mar=c(4,5,3,0.5),cex.lab=1.5,cex.axis=1.5,cex.main=1.5)
plot(x,y,pch="+",col="grey80",xlab="X",ylab="Y",cex=1.2)
lines(xval,predlin,col="darkcyan",lwd=2)
lines(xval,predmars,col="firebrick",lwd=4)
lines(xval,predpoly,col="royalblue3",lwd=3)
legend("topleft",legend=c("Linear","Polynomial
(4)","MARS"),lwd=3,seg.len=3,col=c("darkcyan","royalblue3","firebrick"),bty="n")
```

R code for examples

```
## TITANIC DATA
set.seed(1220)
etitanic$noise.norm <- rnorm(nrow(etitanic))
etitanic$noise.unif <- runif(nrow(etitanic),-2,2)
etitanic$noise.binom <- rbinom(nrow(etitanic),1,0.5)
etitanic$survived2 <- as.factor(ifelse(etitanic$survived==1,"yes","no"))

mars <- earth(survived2~.-
survived,data=etitanic,degree=2,pmethod="exhaustive",glm=list(family=binomial))
summary(mars)

plotmo(mars,pt.col="slategray2",all1=TRUE,degree1=4,degree2=FALSE,lwd=4,col="royalblue4",grid.le
vels=list(sex="male"),do.par=2,mfrow=c(1,2),
main="Gender: Male",xlab="Age",ylab="Probability of survival",cex.lab=1.5,cex.axis=1.5)
abline(h=0.5,lty=3)
plotmo(mars,pt.col="slategray2",all1=TRUE,degree1=4,degree2=FALSE,lwd=4,col="royalblue4",grid.le
vels=list(sex="female"),do.par=FALSE,
main="Gender: Female",xlab="Age",ylab="Probability of survival",cex.lab=1.5,cex.axis=1.5)
abline(h=0.5,lty=3)

evimp(mars)

set.seed(1846)
mars <- earth(survived2~.-
survived,data=etitanic,degree=3,pmethod="cv",glm=list(family=binomial),nfold=5)
summary(mars)
```

R code for examples

```
plotmo(mars,pt.col="slategray2",all1=TRUE,degree1=4,degree2=FALSE,lwd=4,col="royalblue4",grid.levels=list(sex="male",pclass="1st"),do.par=2,mfrow=c(2,3),
main="Male - 1st class",xlab="Age",ylab="Probability of survival",cex.lab=1.2,cex.axis=1)
abline(h=0.5,lty=3)
plotmo(mars,pt.col="slategray2",all1=TRUE,degree1=4,degree2=FALSE,lwd=4,col="royalblue4",grid.levels=list(sex="male",pclass="2nd"),do.par=FALSE,
main="Male - 2nd class",xlab="Age",ylab="Probability of survival",cex.lab=1.2,cex.axis=1)
abline(h=0.5,lty=3)
plotmo(mars,pt.col="slategray2",all1=TRUE,degree1=4,degree2=FALSE,lwd=4,col="royalblue4",grid.levels=list(sex="male",pclass="3rd"),do.par=FALSE,
main="Male - 3rd class",xlab="Age",ylab="Probability of survival",cex.lab=1.2,cex.axis=1)
abline(h=0.5,lty=3)
plotmo(mars,pt.col="slategray2",all1=TRUE,degree1=4,degree2=FALSE,lwd=4,col="royalblue4",grid.levels=list(sex="female",pclass="1st"),do.par=FALSE,
main="Female - 1st class",xlab="Age",ylab="Probability of survival",cex.lab=1.2,cex.axis=1)
abline(h=0.5,lty=3)
plotmo(mars,pt.col="slategray2",all1=TRUE,degree1=4,degree2=FALSE,lwd=4,col="royalblue4",grid.levels=list(sex="female",pclass="2nd"),do.par=FALSE,
main="Female - 2nd class",xlab="Age",ylab="Probability of survival",cex.lab=1.2,cex.axis=1)
abline(h=0.5,lty=3)
plotmo(mars,pt.col="slategray2",all1=TRUE,degree1=4,degree2=FALSE,lwd=4,col="royalblue4",grid.levels=list(sex="female",pclass="3rd"),do.par=FALSE,
main="Female - 3rd class",xlab="Age",ylab="Probability of survival",cex.lab=1.2,cex.axis=1)
abline(h=0.5,lty=3)
```