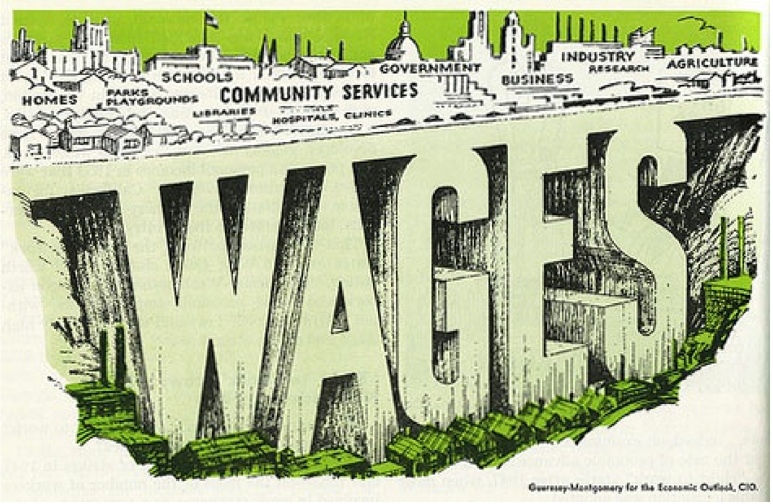
```
understanding how predictors interact
with predicted value is very important.

Therefore: plot!
```

Plotting predictors

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Example: predicting wages



Guernsey-Montgomery for the Economic Outlook, CIO.

Image Credit http://www.cahs-media.org/the-high-cost-of-low-wages

Data from: ISLR package from the book: Introduction to statistical learning

Example: Wage data

```r
library(ISLR); library(ggplot2); library(caret);
data(Wage)
summary(Wage)
```

```
     year           age              sex                    maritl           race     
Min.   :2003   Min.   :18.0   1. Male  :3000   1. Never Married: 648   1. White:2480  
1st Qu.:2004   1st Qu.:33.8   2. Female:   0   2. Married      :2074   2. Black: 293  
Median :2006   Median :42.0                    3. Widowed      :  19   3. Asian: 190  
Mean   :2006   Mean   :42.4                    4. Divorced     : 204   4. Other:  37  
3rd Qu.:2008   3rd Qu.:51.0                    5. Separated    :  55                  
Max.   :2009   Max.   :80.0                                                           

              education                     region               jobclass               health      
1. < HS Grad        :268   2. Middle Atlantic    :3000   1. Industrial :1544   1. <=Good     : 858  
2. HS Grad          :971   1. New England        :   0   2. Information:1456   2. >=Very Good:2142  
3. Some College     :650   3. East North Central:   0                                              
4. College Grad     :685   4. West North Central:   0                                              
5. Advanced Degree:426   5. South Atlantic     :   0                                              
                           6. East South Central:   0                                              
                           (Other)               :   0                                              
```

Get training/test sets

```
inTrain <- createDataPartition(y=Wage$wage,
                              p=0.7, list=FALSE)
training <- Wage[inTrain,]
testing <- Wage[-inTrain,]
dim(training); dim(testing)
```

```
[1] 898  12
```

Create a test set and set it aside: even before you do any exploration. Don’t touch it anymore until you evaluate the model trained on the training set!

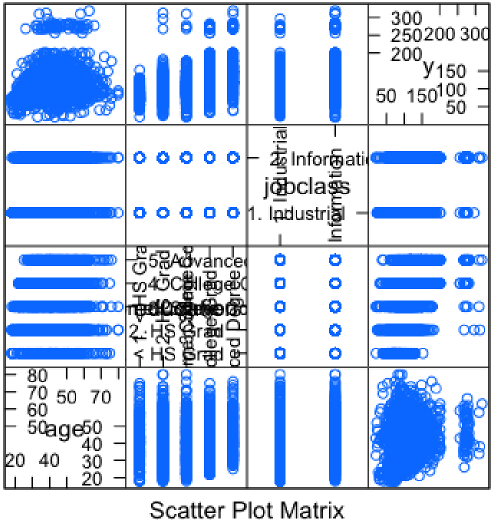
Feature plot (*caret* package)

```
featurePlot(x=training[,c("age","education","jobclass")],
            y = training$wage,
            plot="pairs")
```

Correlate
x=age, education, jobclass
with y=wage

Der Plot zeigt jedoch
Korrelationen von
allen mit allen.

Gesucht: Variablen, die
einen Zusammenhangen mit
y haben.

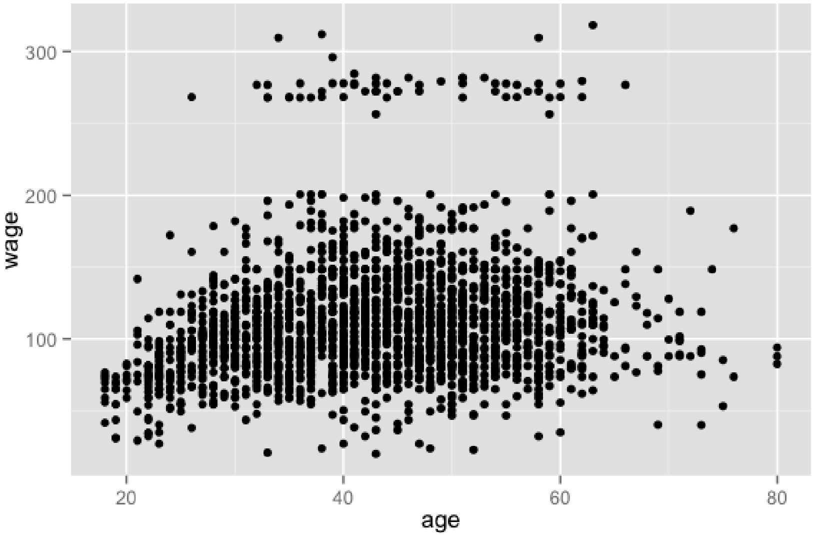


Qplot (*ggplot2* package)

```
qplot(age,wage,data=training)
```

just a scatterplot.

Interessante Wolke dort oben.

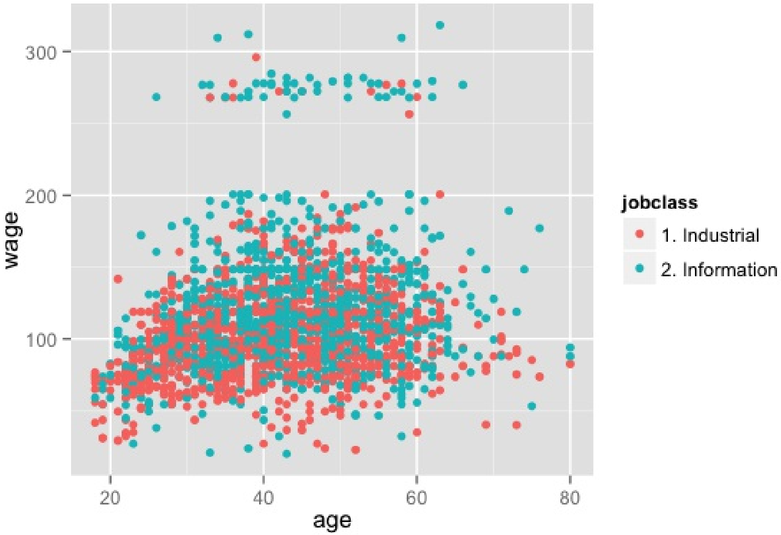


Qplot with color (*ggplot2* package)

```
qplot(age,wage,colour=jobclass,data=training)
```

dasselbe wie oben, aber nun eingefaerbt nach jobclass.

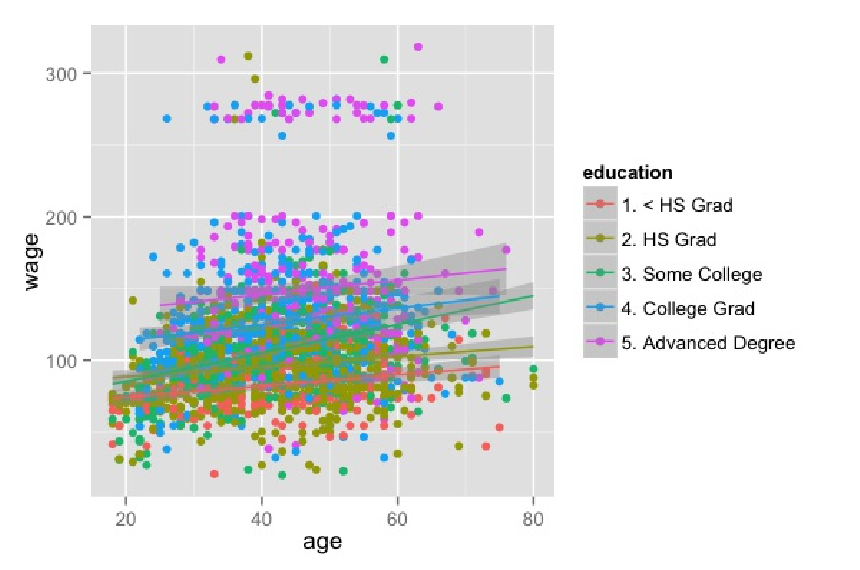
Die Wolke besteht v.a. aus der gruenen Jobclass…



Add regression smoothers (*ggplot2* package)

```
qq <- qplot(age,wage,colour=education,data=training)
qq +  geom_smooth(method='lm',formula=y~x)
```

fitted ein Model fuer jede Stufe von education.



cut2, making factors (*Hmisc* package)

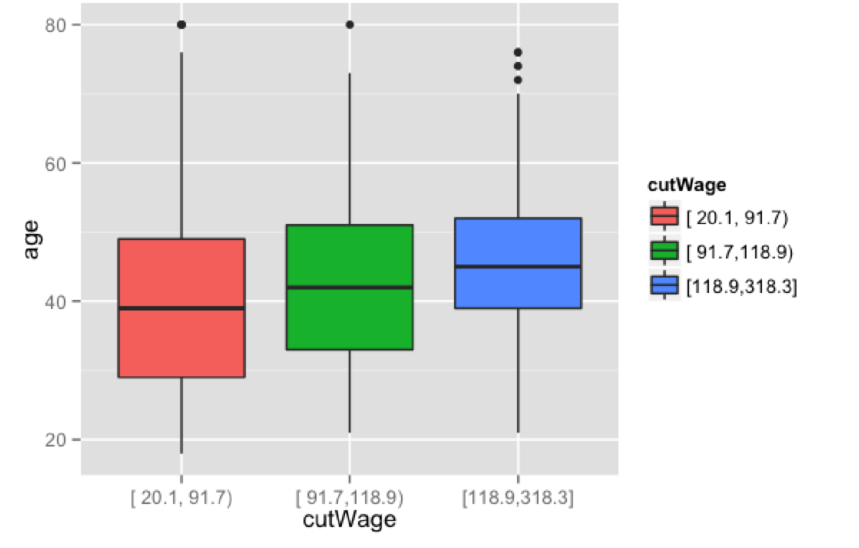
```
cutWage <- cut2(training$wage,g=3)
table(cutWage)
```

create 3 groups of wage (a factor)

```
cutWage
[ 20.1, 91.7) [ 91.7,118.9) [118.9,318.3]
          704           725           673
```

Boxplots with cut2

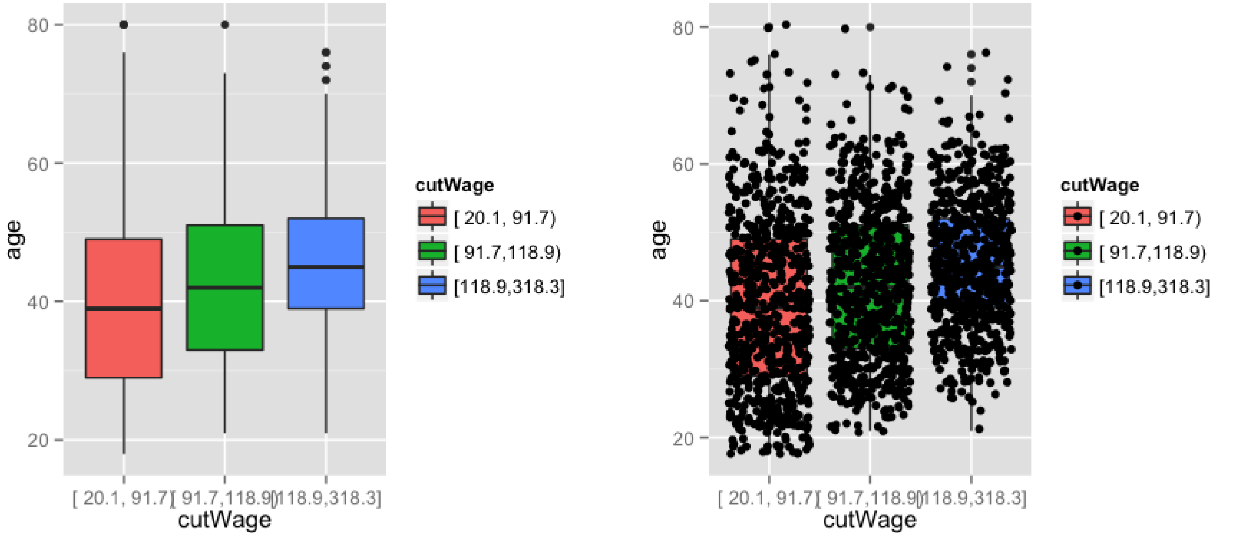
```
p1 <- qplot(cutWage,age, data=training,fill=cutWage,
      geom=c("boxplot"))
p1
```



Boxplots with points overlayed

```
p2 <- qplot(cutWage,age, data=training,fill=cutWage,
      geom=c("boxplot","jitter"))
grid.arrange(p1,p2,ncol=2)
```

Boxplots koennen Beziehung
manchmal deutlicher machen.
Nachteil: man sieht die Anzahl
Datensaetze pro Gruppe nicht.
Loesung: Punkte auch plotten



Tables

aus Faktoren/kategorialen Variablen kann man sich eine Tabelle machen lassen:

```
t1 <- table(cutWage,training$jobclass)
t1
```

```
cutWage         1. Industrial 2. Information
  [ 20.1, 91.7)           437    >       267
  [ 91.7,118.9)           365            360
  [118.9,318.3]           263    <       410
```

```
prop.table(t1,1)
```

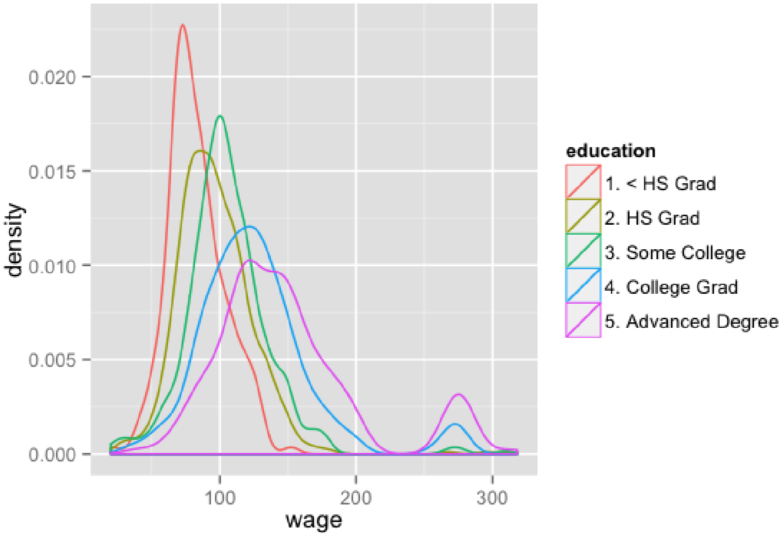
zeigt die Anteile (%) an statt absolute Zahlen:

```
cutWage         1. Industrial 2. Information
  [ 20.1, 91.7)        0.6207         0.3793
  [ 91.7,118.9)        0.5034         0.4966
  [118.9,318.3]        0.3908         0.6092
```

Density plots

fuer kontinuierliche Prediktoren

```
qplot(wage,colour=education,data=training,geom="density")
```



ZB: die Loehne der schlecht Ausgebildeten haeufen sich bei 100,
jene der besser Ausgebildeten bei 150, und dann noch naehe bei 300.

Notes and further reading

- Make your plots only in the training set
  - Don't use the test set for exploration!
- Things you should be looking for
  - Imbalance in outcomes/predictors
  - Outliers
  - Groups of points not explained by a predictor
  - Skewed variables `muss man ev Umformen (log..) falls man Regression verwendet.`
- ggplot2 tutorial
- caret visualizations