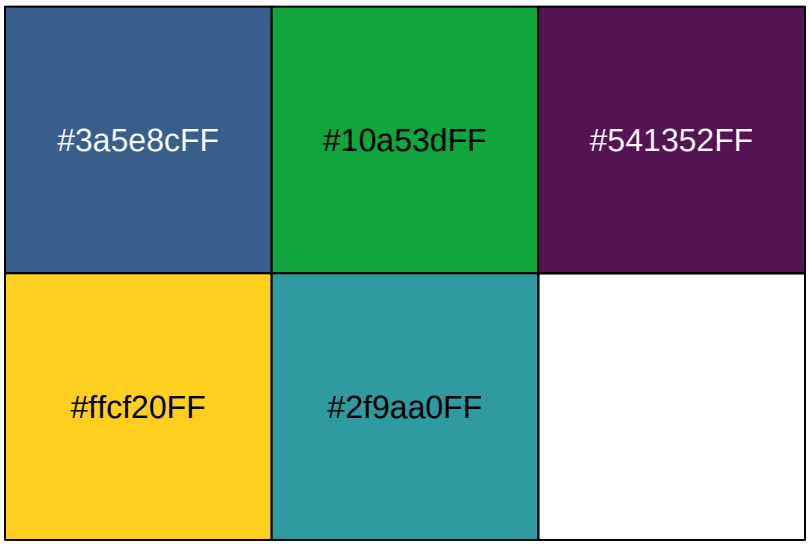
Assignment 3

Elisabeth Fidrmuc

2025-03-27

Data cleaning

```
rm(list=ls())

setwd("C:/Users/elisa/Dropbox/Prediction with Machine Learning/Data-Analysis-3/Assignment 3")
data_dir <- "C:/Users/elisa/Dropbox/Prediction with Machine Learning/Data-Analysis-3/Assignment 3/Data/
output <- "C:/Users/elisa/Dropbox/Prediction with Machine Learning/Data-Analysis-3/Assignment 3/Figures/

# load theme
source("theme_bg.R")
```



```r
source("da_helper_functions.R")


# Packages
library(tidyverse)
library(stargazer)
library(tinytex)
library(dplyr)
library(stringr)
library(janitor)
library(skimr)
library(Hmisc)
library(caret)
```

```r
data <- read.csv(paste0(data_dir, "cs_bisnode_panel.csv"))

# drop important variables with many NAs
data <- data %>%
  select(-c(COGS, finished_prod, net_dom_sales, net_exp_sales, wages)) %>%
  filter(year !=2016)
```

Label engineering

This code prepares the dataset for panel analysis by ensuring consistency across time and firms, engineering key variables, and handling edge cases. First, it fills in all missing combinations of firm ID (`comp_id`) and year, so that each firm appears in every year of the panel—even if the original data was missing for some years—resulting in `NA`s for those entries. It then creates a binary indicator `status_alive` to flag whether a firm is considered active in a given year based on having non-missing, positive sales. Sales values are cleaned by replacing negative entries with 1, then transformed into log scale and converted to millions (`sales_mil_log`) to reduce skewness. The code also calculates the year-over-year change in log sales (`d1_sales_mil_log`) for each firm, which captures growth trends. Finally, it constructs a firm `age` variable and flags newly founded firms (less than or equal to one year old), adjusting their growth measures to zero and handling missing values to ensure consistency. This processing creates a clean and balanced structure ready for modeling firm dynamics over time.

```r
# add all missing year and comp_id combinations -
# originally missing combinations will have NAs in all other columns
data <- data %>%
  complete(year, comp_id)

# generate status_alive; if sales larger than zero and not-NA, then firm is alive
data  <- data %>%
  mutate(status_alive = sales > 0 & !is.na(sales) %>%
           as.numeric(.))


# replace negative sales by 1, log transformation, in millions
data <- data %>%
  mutate(sales = ifelse(sales < 0, 1, sales),
         ln_sales = ifelse(sales > 0, log(sales), 0),
         sales_mil=sales/1000000,
         sales_mil_log = ifelse(sales > 0, log(sales_mil), 0))
```

```r
data <- data %>%
  group_by(comp_id) %>%
  mutate(d1_sales_mil_log = sales_mil_log - Lag(sales_mil_log, 1) ) %>%
  ungroup()


# replace w 0 for new firms + add dummy to capture it
data <- data %>%
  mutate(age = (year - founded_year) %>%
           ifelse(. < 0, 0, .),
         new = as.numeric(age <= 1) %>% #  (age could be 0,1 )
           ifelse(balsheet_notfullyear == 1, 1, .),
         d1_sales_mil_log = ifelse(new == 1, 0, d1_sales_mil_log),
         new = ifelse(is.na(d1_sales_mil_log), 1, new),
         d1_sales_mil_log = ifelse(is.na(d1_sales_mil_log), 0, d1_sales_mil_log))
```

Sample design

In the next step, we filters the dataset to focus on a relevant and consistent sample for analysis. It first restricts the panel to the years 2010 to 2015, ensuring that only data within the main observation window is included. Then, it applies revenue-based filters to exclude extreme cases: firms with sales greater than 10 million euros or less than 1,000 euros (0.001 million) are removed. This step reduces the influence of outliers and firms that may be inactive, misreported, or not economically meaningful. By trimming the dataset this way, the analysis focuses on small to medium-sized firms with more reliable financial activity. Finally, the cleaned dataset is saved as a new CSV file for further modeling and analysis.

Furthermore, this code creates a new variable, `growth_past`, which captures each firm's sales growth from 2011 to 2012. It filters the dataset to keep only those two years, reshapes the data so that 2011 and 2012 sales appear in separate columns, and then calculates the relative growth rate. The resulting growth variable is merged back into the main dataset and can be used as a predictor, helping to capture a firm's past performance without leaking information from the future growth period.

```r
# look at cross section
data <- data %>%
  filter(year >= 2010, # focus on panel from 2010 to 2015
         year <= 2015) %>%
  # look at firms below 10m euro revenues and above 1000 euros
  filter(!(sales_mil > 10)) %>%
  filter(!(sales_mil < 0.001))

# Step 1: Get 2010 and 2012 sales per firm
past_growth <- data %>%
  filter(year %in% c(2011, 2012)) %>%
  select(comp_id, year, sales_mil) %>%
  pivot_wider(names_from = year, values_from = sales_mil, names_prefix = "sales_") %>%
  mutate(
    growth_past = (sales_2012 - sales_2011) / sales_2011,
    growth_past = ifelse(is.infinite(growth_past) | is.nan(growth_past), NA, growth_past)  # handle div
  )

# Step 2: Merge this past growth back into your data
data <- data %>%
```

```
  left_join(past_growth %>% select(comp_id, growth_past), by = "comp_id")

write_csv(data,paste0(data_dir,"work5.csv"))
```

Target design

In this project, we define the target variable fast growth as being in the top 10% of firms based on sales growth between 2012 and 2014. This definition captures sustained expansion in a firm's market activity over a two-year period, which is a meaningful indicator of performance from a corporate finance perspective. Sales (or revenue) is a fundamental metric used to assess a firm's size, operating scale, and top-line performance. It is less volatile than profit measures and less susceptible to short-term accounting effects, making it a more reliable indicator of growth potential. Especially for younger or high-growth firms, sales is a clearer signal of momentum than profit, which may be negative during periods of strategic investment.

We chose a two-year window (2012–2014) to identify fast-growing firms rather than just one year to smooth out temporary fluctuations and avoid capturing growth due to one-off events or noise. Sustained growth over multiple periods is more informative from a strategic and financial planning perspective and aligns better with the interests of investors, creditors, and corporate managers who are concerned with long-term value creation. In corporate finance, growth sustainability and scale expansion are often linked to improved valuation multiples and increased access to external financing. Firms with consistently rising sales are more likely to secure debt or equity funding, attract talent, and expand into new markets.

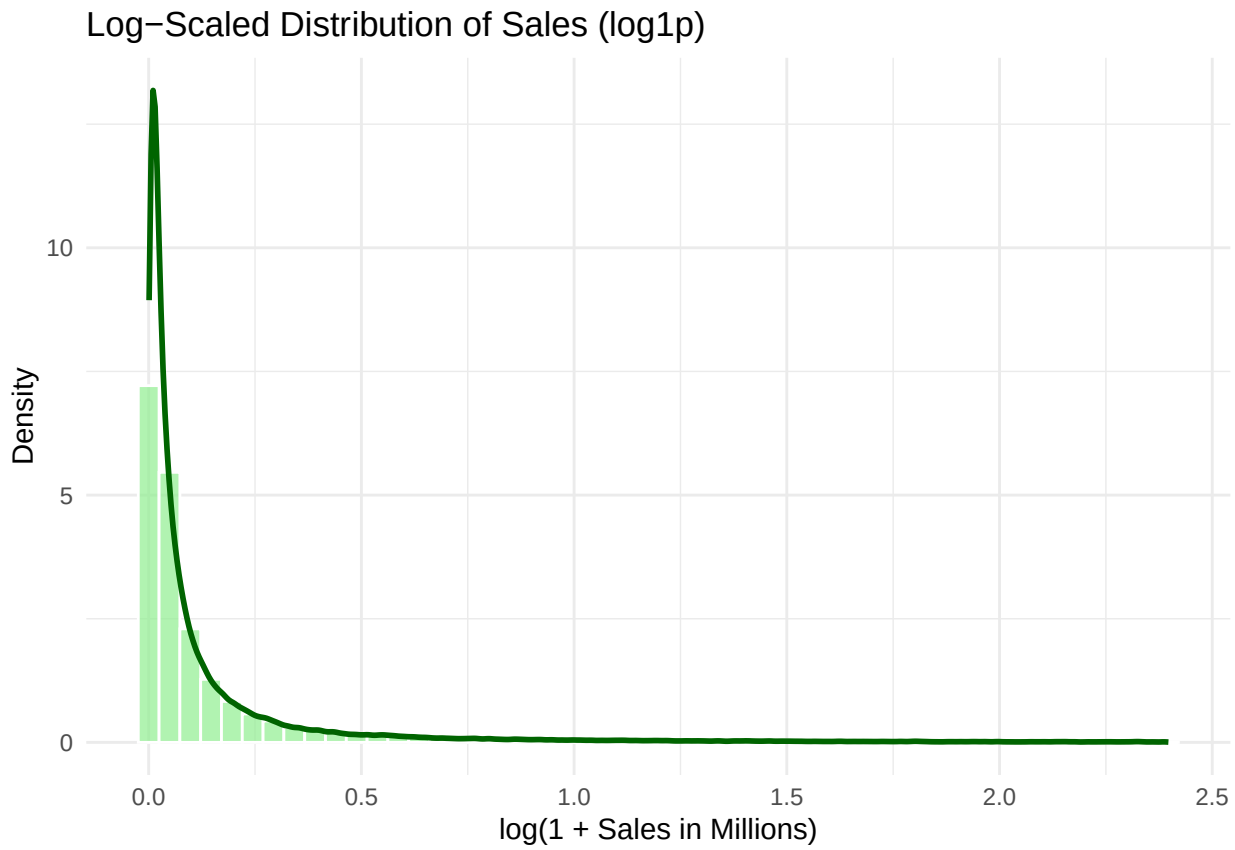
Alternative approaches were considered, such as using profit growth, total asset growth, or employment growth. Profit-based measures can be misleading in early-stage or high-growth firms, where reinvestment strategies or high fixed costs can suppress net income despite strong performance. Asset growth may not reflect real business expansion, especially in service-based firms with low physical capital needs. Employment growth is another commonly used proxy for firm expansion, as growing firms often hire more workers. However, it may lag behind revenue growth or reflect changes in organizational structure rather than true market growth. Additionally, employment data can be more affected by firm-specific HR strategies or outsourcing practices. In contrast, using sales growth as a target combines relevance, availability, and financial interpretability, making it the most appropriate and consistent choice for identifying high-potential firms in this dataset.

This code chunk defines the target variable `fast_growth` based on firms' sales performance between 2012 and 2014. It first filters the dataset to keep only observations from 2012 and 2014, selects the relevant variables (`comp_id`, `year`, and `sales_mil`), and reshapes the data so each firm has its 2012 and 2014 sales on the same row. It then computes the relative sales growth and assigns a value of 1 to firms in the top 20% of the growth distribution (i.e., fast-growing firms), and 0 otherwise. The result is merged back into the main `data` frame, so that each firm is tagged with a binary `fast_growth` label. Finally, I restrict the sample to the cross-section of firms that are alive in 2012.

```
# Calculate sales growth between 2012 and 2014
data <- read.csv(paste0(data_dir,"work5.csv"))

Hmisc::describe(data$sales_mil)
```

```
## data$sales_mil
##        n  missing distinct     Info     Mean  pMedian      Gmd      .05
##   128355        0    58561        1   0.2444  0.07332   0.3935 0.002929
##      .10      .25      .50      .75      .90      .95
## 0.005539 0.015872 0.045919 0.143017 0.447864 0.991129
##
## lowest : 0.001      0.0010037  0.00100741 0.00101111 0.00101481
## highest: 9.93556    9.96352    9.96393    9.96448    9.98931
```

```r
ggplot(data, aes(x = log1p(sales_mil))) +  # log1p handles zero safely
  geom_histogram(aes(y = ..density..), bins = 50, fill = "lightgreen", color = "white", alpha = 0.7) +
  geom_density(color = "darkgreen", size = 1) +
  labs(
    title = "Log-Scaled Distribution of Sales (log1p)",
    x = "log(1 + Sales in Millions)",
    y = "Density"
  ) +
  theme_minimal()
```




```r
sales_growth <- data %>%
  filter(year %in% c(2012, 2014)) %>%
  select(comp_id, year, sales_mil) %>%
  pivot_wider(names_from = year, values_from = sales_mil, names_prefix = "sales_") %>%
  mutate(
    growth = (sales_2014 - sales_2012) / sales_2012,
    fast_growth = ifelse(
      !is.na(growth) & growth > quantile(growth, 0.8, na.rm = TRUE), 1, 0
    )
  )

# Merge back to main data
data <- data %>%
  left_join(sales_growth %>% select(comp_id, fast_growth), by = "comp_id")
table(data$fast_growth)
```

```
##
##      0      1
## 101353  17272
```

```
sum(is.na(data$fast_growth))
```

```
## [1] 9730
```

```
# drop if target is missing
data <- data %>% filter(!is.na(fast_growth))

# focus on cross-section in 2012
data <- data %>%
  filter((year==2012) & (status_alive==1))
```

The figure shows the distribution of firm sales (in millions of euros), transformed using the natural logarithm with a log1p scale to handle zero values. The distribution is heavily right-skewed, indicating that the majority of firms in the dataset have relatively low sales, with a dense concentration near zero. This suggests that most firms are micro or small enterprises. Despite the log transformation, there is still a long tail to the right, representing a small number of firms with significantly higher sales. The skewness justifies the use of log transformation to stabilize variance and make the data more suitable for modeling.

Feature engineering

This section performs extensive feature engineering and data cleaning to prepare the dataset for modeling. First, it recodes industry classification (`ind2`) into broader industry categories (`ind2_cat`) to simplify analysis. It then creates new firm characteristics, including squared age, a binary foreign management indicator, and region and gender factors. Financial variables are also prepared: problematic asset values are flagged and replaced with zero if negative, and total assets are computed by summing different asset components. Key profit and balance sheet items are normalized—profit and loss elements by sales, and balance sheet items by total assets—allowing for meaningful comparisons across firms of different sizes. Flags are generated for unrealistic values (e.g., ratios above 1 or below -1), and extreme values are winsorized to prevent them from distorting the models. Flags with no variation are dropped. Additional imputation handles missing or extreme values for CEO age and average labor input, with outliers capped and missing values replaced by the mean. Finally, factor variables are created for industry category, urban location and the target fast growth, converting them into categorical variables that are ready for use in classification models. This comprehensive preparation ensures the data is clean, interpretable, and suitable for robust predictive modeling.

```
# change some industry category codes
data <- data %>%
  mutate(ind2_cat = ind2 %>%
           ifelse(. > 56, 60, .)  %>%
           ifelse(. < 26, 20, .) %>%
           ifelse(. < 55 & . > 35, 40, .) %>%
           ifelse(. == 31, 30, .) %>%
           ifelse(is.na(.), 99, .)
           )

table(data$ind2_cat)
```

```
##
##    20    26    27    28    29    30    32    33    40    55    56    60    99
##    50  1080   660  1954   287   168   148  1944   208  2266 12711   242     5
```

```r
# Firm characteristics
data <- data %>%
  mutate(age2 = age^2,
         foreign_management = as.numeric(foreign >= 0.5),
         gender_m = factor(gender, levels = c("female", "male", "mix")),
         m_region_loc = factor(region_m, levels = c("Central", "East", "West")))

# Financial variables

# assets can't be negative. Change them to 0 and add a flag.
data <-data  %>%
  mutate(flag_asset_problem=ifelse(intang_assets<0 | curr_assets<0 | fixed_assets<0,1,0  ))
table(data$flag_asset_problem)
```

```
##
##     0     1
## 21700    13
```

```r
data <- data %>%
  mutate(intang_assets = ifelse(intang_assets < 0, 0, intang_assets),
         curr_assets = ifelse(curr_assets < 0, 0, curr_assets),
         fixed_assets = ifelse(fixed_assets < 0, 0, fixed_assets))

# generate total assets
data <- data %>%
  mutate(total_assets_bs = intang_assets + curr_assets + fixed_assets)
summary(data$total_assets_bs)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.      NA's
##         0      6467     23830    252589     93463 104683276        10
```

```r
pl_names <- c("extra_exp","extra_inc",  "extra_profit_loss", "inc_bef_tax" ,"inventories",
              "material_exp", "profit_loss_year", "personnel_exp")
bs_names <- c("intang_assets", "curr_liab", "fixed_assets", "liq_assets", "curr_assets",
              "share_eq", "subscribed_cap", "tang_assets" )

# divide all pl_names elements by sales and create new column for it
data <- data %>%
  mutate_at(vars(pl_names), funs("pl"=./sales))

# divide all bs_names elements by total_assets_bs and create new column for it
data <- data %>%
  mutate_at(vars(bs_names), funs("bs"=ifelse(total_assets_bs == 0, 0, ./total_assets_bs)))


# flags and winsorizing tails
# Variables that represent accounting items that cannot be negative (e.g. materials)
zero <-  c("extra_exp_pl", "extra_inc_pl", "inventories_pl", "material_exp_pl", "personnel_exp_pl",
           "curr_liab_bs", "fixed_assets_bs", "liq_assets_bs", "curr_assets_bs", "subscribed_cap_bs",
           "intang_assets_bs")

data <- data %>%
```

```
  mutate_at(vars(zero), funs("flag_high"= as.numeric(.> 1))) %>%
  mutate_at(vars(zero), funs(ifelse(.> 1, 1, .))) %>%
  mutate_at(vars(zero), funs("flag_error"= as.numeric(.< 0))) %>%
  mutate_at(vars(zero), funs(ifelse(.< 0, 0, .)))


# for vars that could be any, but are mostly between -1 and 1
any <-  c("extra_profit_loss_pl", "inc_bef_tax_pl", "profit_loss_year_pl", "share_eq_bs")

data <- data %>%
  mutate_at(vars(any), funs("flag_low"= as.numeric(.< -1))) %>%
  mutate_at(vars(any), funs(ifelse(.< -1, -1, .))) %>%
  mutate_at(vars(any), funs("flag_high"= as.numeric(.> 1))) %>%
  mutate_at(vars(any), funs(ifelse(.> 1, 1, .))) %>%
  mutate_at(vars(any), funs("flag_zero"= as.numeric(.== 0))) %>%
  mutate_at(vars(any), funs("quad"= .^2))


# dropping flags with no variation
variances<- data %>%
  select(contains("flag")) %>%
  apply(2, var, na.rm = TRUE) == 0

data <- data %>%
  select(-one_of(names(variances)[variances]))


# additional
# including some imputation

# CEO age
data <- data %>%
  mutate(ceo_age = year-birth_year,
         flag_low_ceo_age = as.numeric(ceo_age < 25 & !is.na(ceo_age)),
         flag_high_ceo_age = as.numeric(ceo_age > 75 & !is.na(ceo_age)),
         flag_miss_ceo_age = as.numeric(is.na(ceo_age)))

data <- data %>%
  mutate(ceo_age = ifelse(ceo_age < 25, 25, ceo_age) %>%
           ifelse(. > 75, 75, .) %>%
           ifelse(is.na(.), mean(., na.rm = TRUE), .),
         ceo_young = as.numeric(ceo_age < 40))

# number emp, very noisy measure
data <- data %>%
  mutate(labor_avg_mod = ifelse(is.na(labor_avg), mean(labor_avg, na.rm = TRUE), labor_avg),
         flag_miss_labor_avg = as.numeric(is.na(labor_avg)))

summary(data$labor_avg)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  0.0833  0.0972  0.2292  0.6217  0.5139 42.1181    3154
```

```
summary(data$labor_avg_mod)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.08333  0.12500  0.28472  0.62169  0.62169 42.11806
```

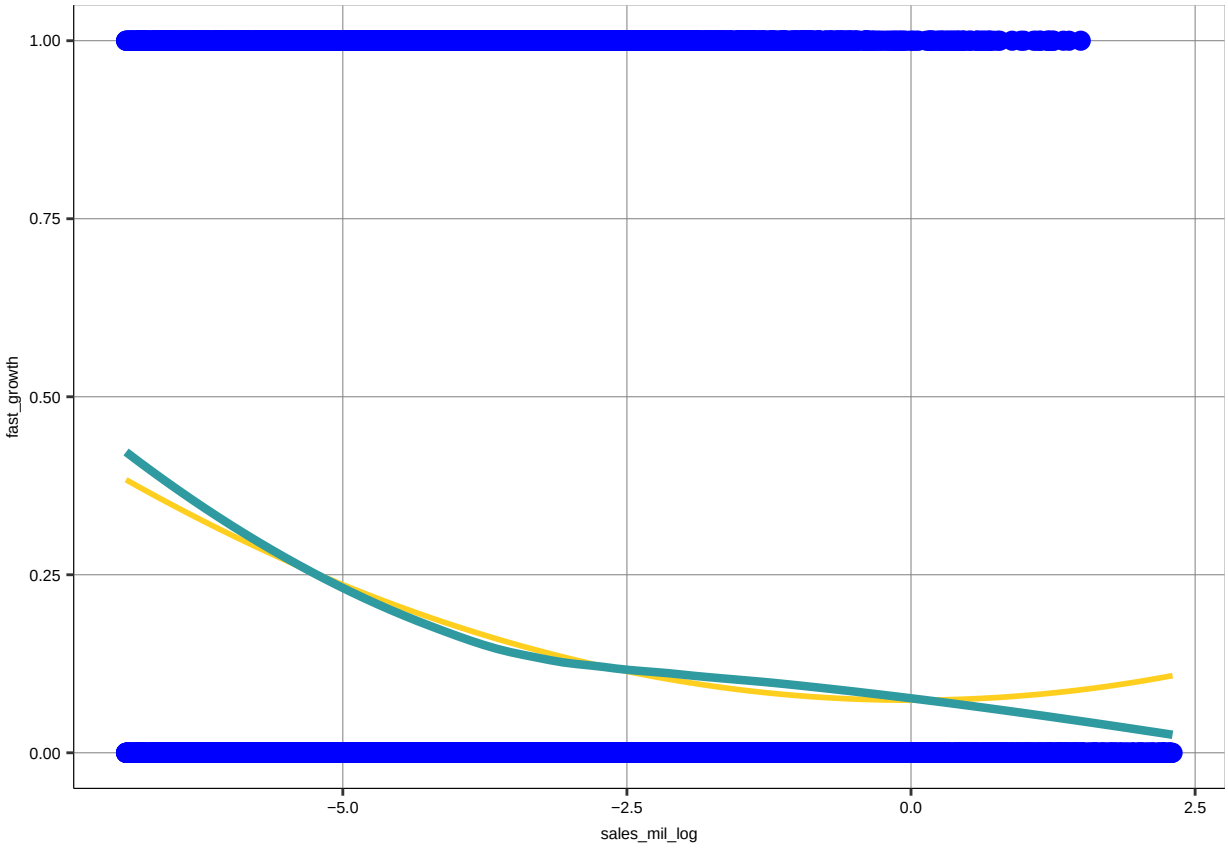
```
data <- data %>%
  select(-labor_avg)

# create factors
data <- data %>%
  mutate(urban_m = factor(urban_m, levels = c(1,2,3)),
         ind2_cat = factor(ind2_cat, levels = sort(unique(data$ind2_cat))))

data <- data %>%
  mutate(fast_growth_f = factor(fast_growth, levels = c(0,1)) %>%
           recode(., `0` = 'no_fast_growth', `1` = "fast_growth"))

table(data$fast_growth_f)
```

```
##
## no_fast_growth    fast_growth
##          18349           3374
```

Sales

This code explores the relationship between firm sales, sales growth, and the probability of being classified as a fast-growing firm. It begins by creating a squared term for log-transformed sales to capture potential non-linear effects in a linear probability model, which is then visualized using both polynomial and loess smoothing. A separate plot investigates how annual sales growth (measured as the log difference in sales) relates to fast growth, again using loess smoothing to capture the trend. To address extreme values, the code flags and winsorizes unusually large or small growth rates, capping them between -1.5 and 1.5, and creates a squared term of the cleaned variable for modeling. Observations with missing values in key variables are dropped to ensure model readiness, and unused factor levels are removed. The code concludes with additional visualizations comparing the original and winsorized growth metrics, illustrating the effect of the transformation and preparing the data for predictive modeling.

```
data <- data %>%
  mutate(sales_mil_log_sq=sales_mil_log^2)


ggplot(data = data, aes(x=sales_mil_log, y=as.numeric(fast_growth))) +
  geom_point(size=2,  shape=20, stroke=2, fill="blue", color="blue") +
  geom_smooth(method = "lm", formula = y ~ poly(x,2), color=color[4], se = F, size=1)+
  geom_smooth(method="loess", se=F, colour=color[5], size=1.5, span=0.9) +
  labs(x = "sales_mil_log",y = "fast_growth") +
  theme_bg()
```



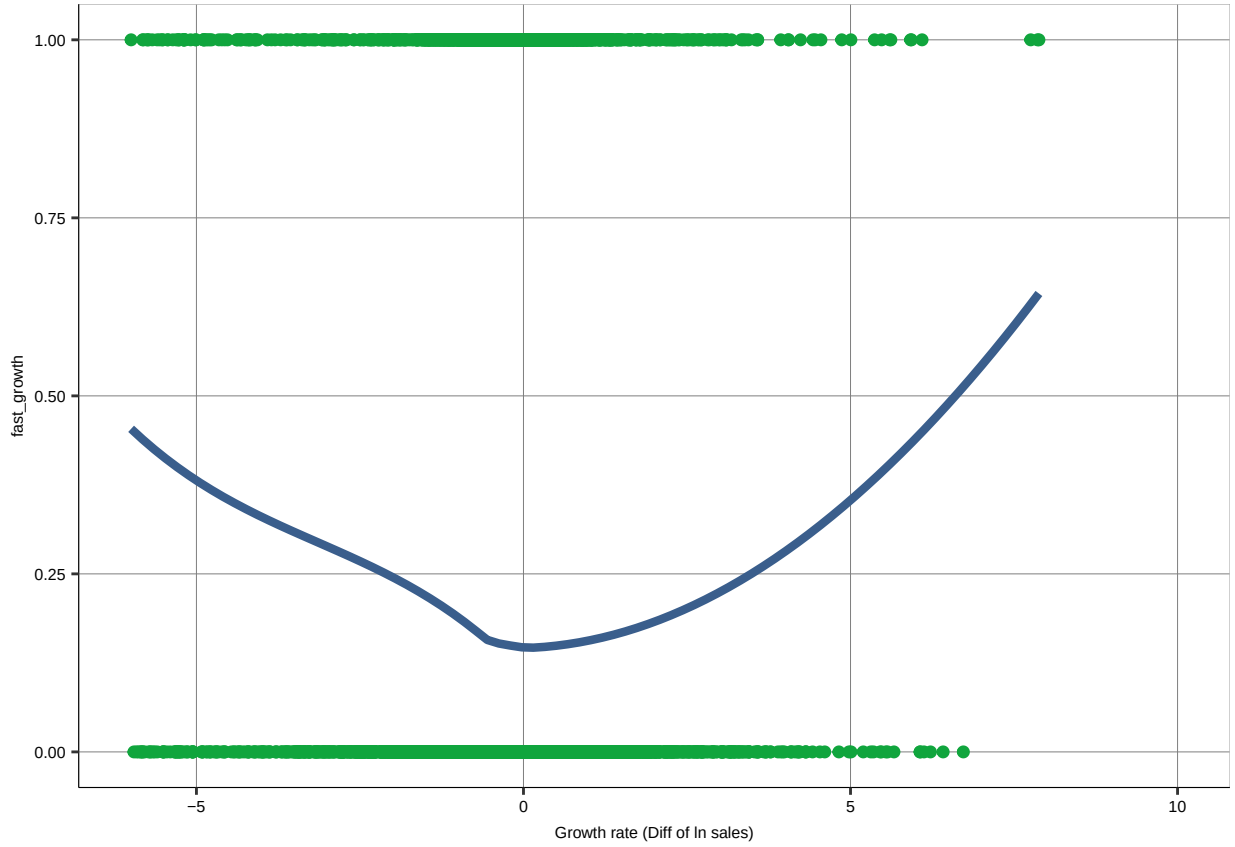

```
ols_s <- lm(fast_growth~sales_mil_log+sales_mil_log_sq,
                data = data)
summary(ols_s)
```

```
##
## Call:
## lm(formula = fast_growth ~ sales_mil_log + sales_mil_log_sq,
##     data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38343 -0.17284 -0.12100 -0.08168  0.92614
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      7.386e-02  5.832e-03  12.664   <2e-16 ***
## sales_mil_log    4.613e-05  3.631e-03   0.013     0.99
## sales_mil_log_sq 6.494e-03  5.876e-04  11.052   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3557 on 21720 degrees of freedom
## Multiple R-squared:  0.03586,    Adjusted R-squared:  0.03577
## F-statistic: 403.9 on 2 and 21720 DF,  p-value: < 2.2e-16
```

```
# lowess
Hmisc::describe(data$d1_sales_mil_log) # no missing
```

```
## data$d1_sales_mil_log
##          n    missing   distinct       Info       Mean    pMedian        Gmd
##      21723          0      15927      0.982   -0.07768 -0.0002419     0.6512
##        .05        .10        .25        .50        .75        .90        .95
##    -1.1546    -0.5923    -0.1585     0.0000     0.1034     0.4098     0.7645
##
## lowest : -6.88941 -6.84322 -6.8363  -6.82942 -6.8226
## highest: 6.41602  6.72521  7.75534  7.85953  7.8803
```

```
d1sale_1<-ggplot(data = data, aes(x=d1_sales_mil_log, y=as.numeric(fast_growth))) +
  geom_point(size=0.1,  shape=20, stroke=2, fill=color[2], color=color[2]) +
  geom_smooth(method="loess", se=F, colour=color[1], size=1.5, span=0.9) +
 labs(x = "Growth rate (Diff of ln sales)",y = "fast_growth") +
 theme_bg() +
 scale_x_continuous(limits = c(-6,10), breaks = seq(-5,10, 5))
d1sale_1
```



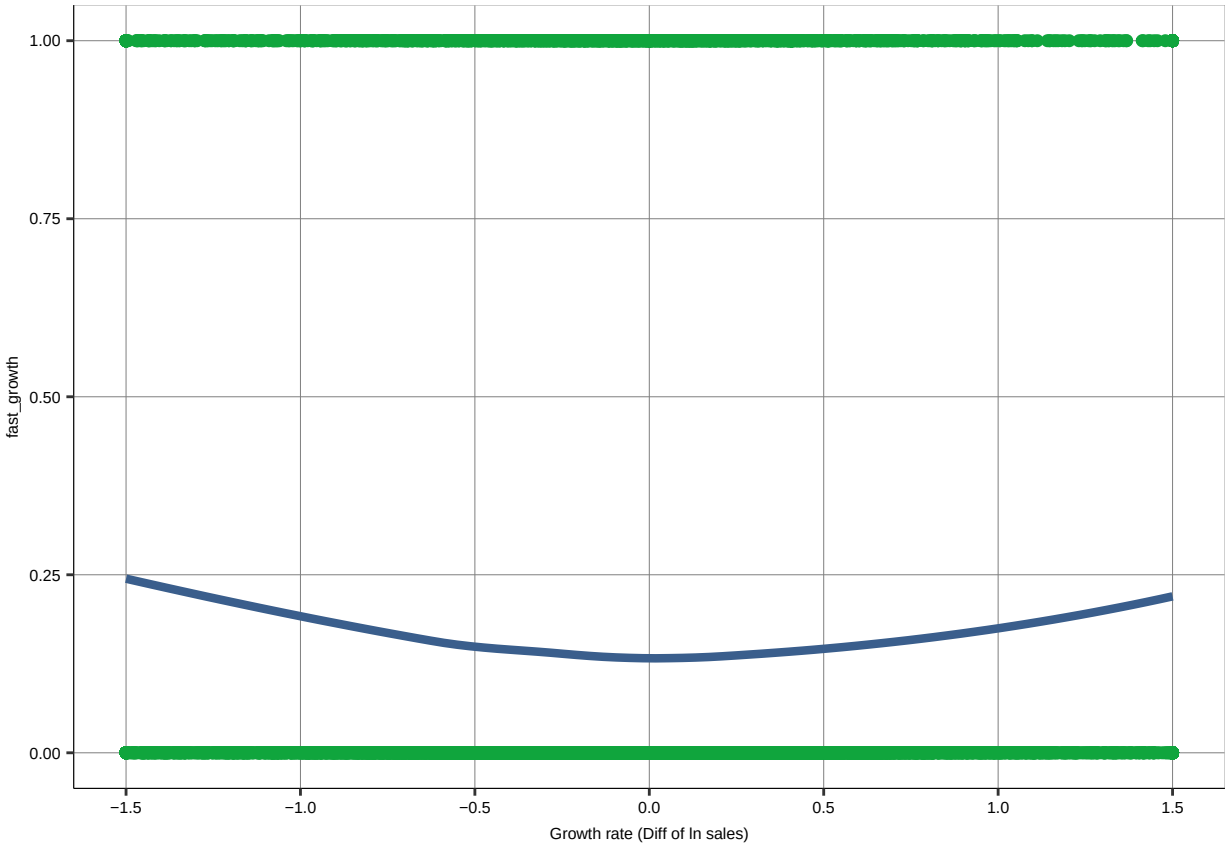

```
save_fig("ch17-extra-1", output, "small")
```

```
## pdf
##   2
```

```
# generate variables -----------------------------------------------------

data <- data %>%
  mutate(flag_low_d1_sales_mil_log = ifelse(d1_sales_mil_log < -1.5, 1, 0),
         flag_high_d1_sales_mil_log = ifelse(d1_sales_mil_log > 1.5, 1, 0),
         d1_sales_mil_log_mod = ifelse(d1_sales_mil_log < -1.5, -1.5,
                                       ifelse(d1_sales_mil_log > 1.5, 1.5, d1_sales_mil_log)),
         d1_sales_mil_log_mod_sq = d1_sales_mil_log_mod^2
         )

# no more imputation, drop obs if key vars missing
data <- data %>%
  filter(!is.na(liq_assets_bs),!is.na(foreign), !is.na(ind))

# drop missing
data <- data %>%
  filter(!is.na(age),!is.na(foreign), !is.na(material_exp_pl), !is.na(m_region_loc))
Hmisc::describe(data$age)
```

```
## data$age
##        n  missing distinct     Info     Mean  pMedian      Gmd      .05
##    19036        0       31    0.996    8.702      8.5    7.827        1
##      .10      .25      .50      .75      .90      .95
##        1        2        7       15       19       21
##
## lowest :  0  1  2  3  4, highest: 26 28 29 32 34
```

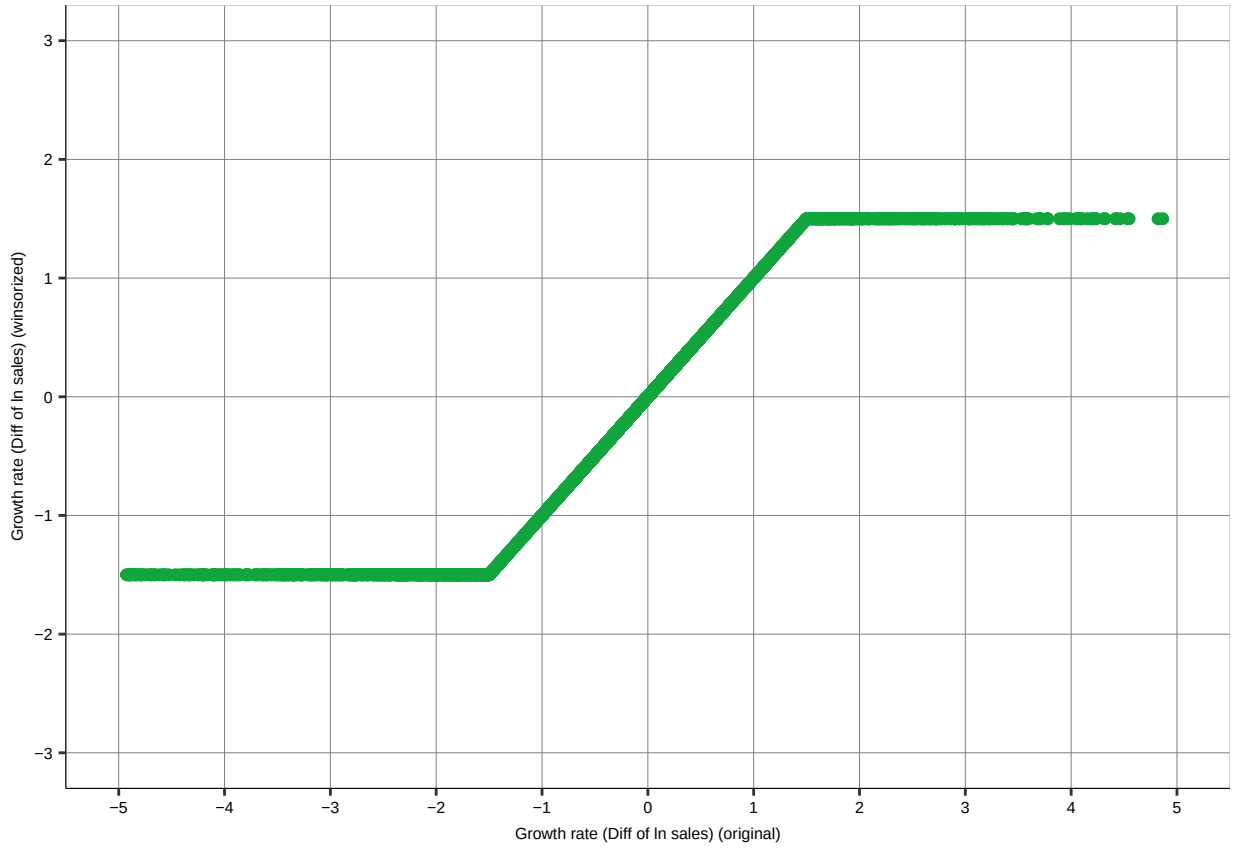
```
# drop unused factor levels
data <- data %>%
  mutate_at(vars(colnames(data)[sapply(data, is.factor)]), funs(fct_drop))

d1sale_2<-ggplot(data = data, aes(x=d1_sales_mil_log_mod, y=as.numeric(fast_growth))) +
  geom_point(size=0.1,  shape=20, stroke=2, fill=color[2], color=color[2]) +
  geom_smooth(method="loess", se=F, colour=color[1], size=1.5, span=0.9) +
  labs(x = "Growth rate (Diff of ln sales)",y = "fast_growth") +
  theme_bg() +
  scale_x_continuous(limits = c(-1.5,1.5), breaks = seq(-1.5,1.5, 0.5))
d1sale_2
```




```
save_fig("ch17-extra-2", output, "small")
```

```
## pdf
##   2
```

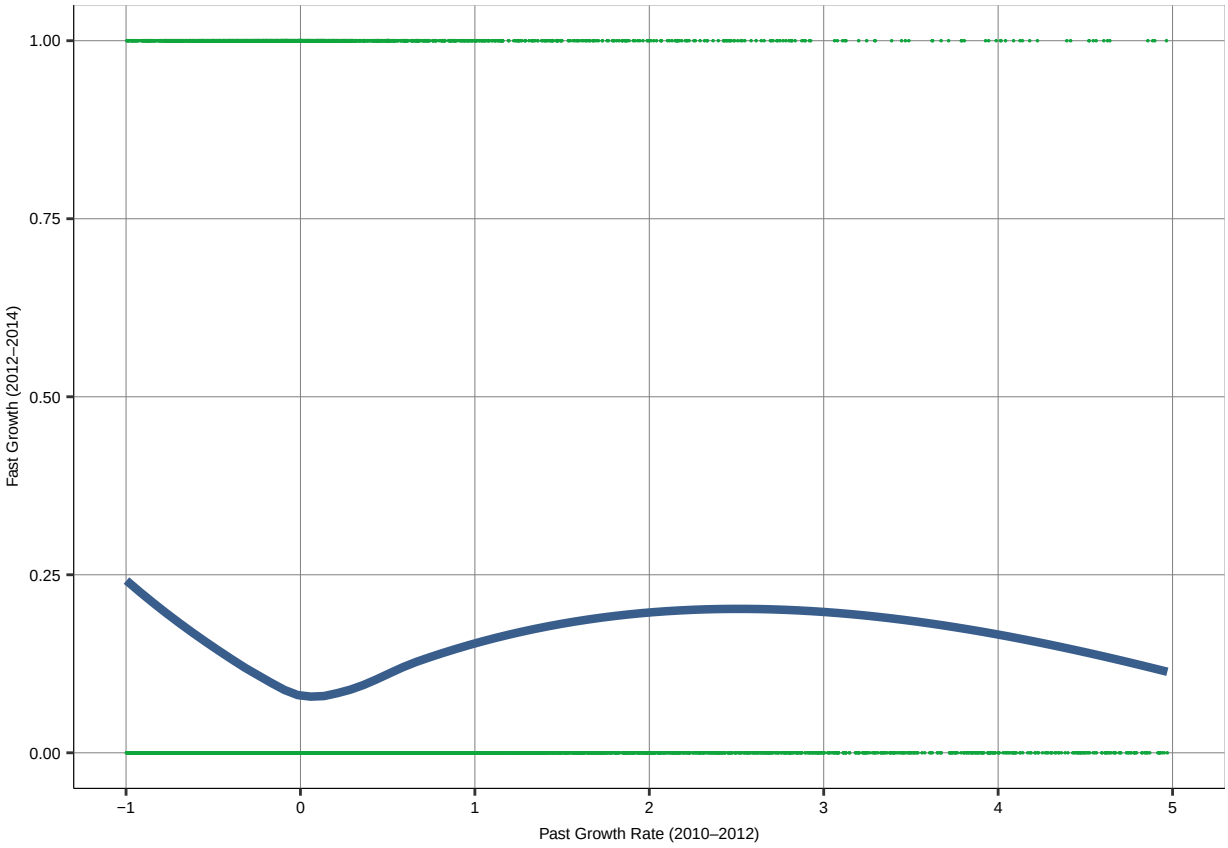
```
d1sale_3<-ggplot(data = data, aes(x=d1_sales_mil_log, y=d1_sales_mil_log_mod)) +
  geom_point(size=0.1,  shape=20, stroke=2, fill=color[2], color=color[2]) +
  labs(x = "Growth rate (Diff of ln sales) (original)",y = "Growth rate (Diff of ln sales) (winsorized)
  theme_bg() +
  scale_x_continuous(limits = c(-5,5), breaks = seq(-5,5, 1)) +
scale_y_continuous(limits = c(-3,3), breaks = seq(-3,3, 1))
d1sale_3
```




```
save_fig("ch17-extra-3", output, "small")
```

```
## pdf
##   2
```

```
# past growth
growth_past_plot <- ggplot(data = data, aes(x = growth_past, y = as.numeric(fast_growth))) +
  geom_point(size = 0.1, shape = 20, color = color[2]) +
  geom_smooth(method = "loess", se = FALSE, color = color[1], size = 1.5, span = 0.9) +
  labs(x = "Past Growth Rate (2010-2012)", y = "Fast Growth (2012-2014)") +
  theme_bg() +
  scale_x_continuous(limits = c(-1, 5), breaks = seq(-1, 5, 1))  # adjust limits as needed

# Show and save
growth_past_plot
```




```
save_fig("growth-past-vs-fastgrowth", output, "small")
```

```
## pdf
##   2
```

The plots explore the relationship between firm characteristics (especially sales and growth) and the likelihood of being classified as a fast-growing firm. The first plot shows that the probability of being labeled fast-growing declines with log-transformed sales: smaller firms are more likely to experience fast growth, while larger firms rarely do. This reflects a typical pattern in firm dynamics—smaller firms have more room to grow proportionally, while larger firms face diminishing returns to scale. The curve is smooth and downward-sloping, with both polynomial and loess fits capturing the non-linear decline.

The second and third plots examine current sales growth (as the log-difference in sales) and its winsorized version. They reveal a U-shaped relationship: both very low and very high short-term sales growth rates are associated with a higher probability of fast growth. This might suggest that some firms bounce back sharply after a dip (catch-up growth), while others are already on a fast upward trajectory. However, because fast growth is defined over a two-year period, including recent sales growth as a predictor risks leakage. The fourth plot illustrates the winsorization process, showing how extreme values are capped to prevent distortion. Finally, the fifth plot shows the relationship between past growth (2010–2012) and future fast growth (2012–2014). This relationship is mildly non-linear and less clear-cut, with a slight hump-shaped pattern: very low or very high past growth is less predictive of future fast growth, while moderate growth may be more indicative of sustained upward momentum. This suggests that while past growth carries some predictive power, it's not a dominant factor on its own.

Finally, I save the data.

```r
write_csv(data,paste0(data_dir,"bisnode_firms_clean.csv"))
write_rds(data,paste0(data_dir,"bisnode_firms_clean.rds"))
```