Read someone else's code

Silvie Cinková

2025-08-09

```
  library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.4
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
  billionaires_df <- read_tsv("../datasets_ATRIUM/billionaires_combined.tsv")
```

```
Rows: 28986 Columns: 21
-- Column specification --------------------------------------------------------
Delimiter: "\t"
chr (17): person, name.x, state, headquarters, source, industry, gender, las...
dbl  (4): time, daily_income, age, birth_comb

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
  billionaires_2020 <- billionaires_df %>%
    filter(time == 2020) %>% select(daily_income, person, world_6region)
```

1 The task performed by the code

The data set compiles charts of world's billionaires between 2002 and 2020. This means that the same person can occur more than one time, but their details may differ. Plot billionaires in each world region (column `world_6region`). Use boxplots to plot the distribution of `daily_income` among all billionaires in the given world region and text labels with values from the column `person` for outliers. You may have to plot each region separately, that is, not break one plot into facets but run the/a plotting script individually for each world region.

Solution

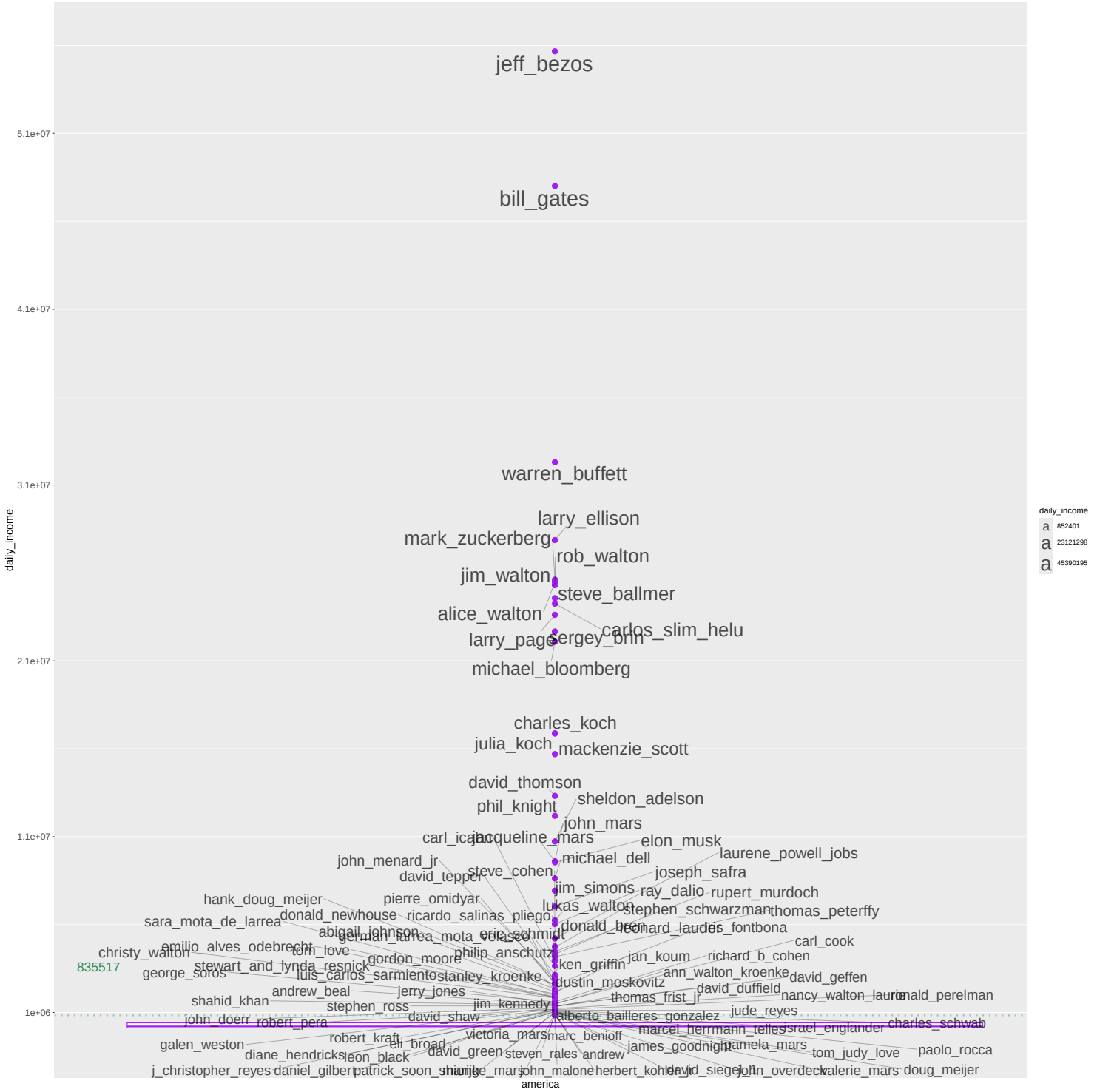
```
library(ggrepel)
billionaires_outliers <- billionaires_2020 %>%
  group_by(world_6region) %>%
  mutate(Q3 = quantile(daily_income, 0.75),
         IQR_col = IQR(daily_income),
         outliers_above = Q3 + 1.5 * IQR_col) %>%
  mutate(is_outlier = if_else(condition = daily_income > outliers_above,
                              true = TRUE,
                              false = FALSE)) %>%
  arrange(desc(daily_income), world_6region)
world_6region_vec <- distinct(billionaires_outliers, world_6region) %>%
  arrange(world_6region) %>%
  pull()
outliers_above <- distinct(billionaires_outliers, world_6region, outliers_above) %>%
  arrange(world_6region) %>%
  pull()

for (i in seq_along(world_6region_vec)) {
  all_oneregion <- billionaires_outliers %>%
    filter(world_6region == world_6region_vec[i])
  outliers_oneregion <- all_oneregion %>%
    filter(is_outlier == TRUE)
  y_axis_offset_for_outlier_label <- range(all_oneregion$daily_income) %>%
    diff()
  y_axis_offset_for_outlier_label <- y_axis_offset_for_outlier_label * 0.05
  if (nrow(outliers_oneregion) == 0){
    cat(world_6region_vec[i], "has no outliers. I will plot all names.\n")
  p <- ggplot() +
    geom_boxplot(data = all_oneregion,
                 mapping = aes(y = daily_income, x = 1),
                 color = "purple")   +
    geom_text_repel(mapping = aes(y = daily_income,
                                  x = 1,
                                  label = person,
                              size = daily_income),
                    data = all_oneregion,
                    max.overlaps = 100,
```

```
                    force = 7,
                    alpha = 0.7,
                    segment.alpha = 0.2,
                    segment.size = 0.1) +
  scale_x_continuous(breaks = NULL,
                     name = world_6region_vec[i]) +
  scale_y_continuous(breaks = seq(from = 10^6,
                                  to = ceiling(round(
                                  max(all_oneregion$daily_income) * 10^(-6)) * 10^6),
                                  by = 10^7),
                     labels = as.character(
                                       seq(from = 10^6,
                                           to = ceiling(round(max(all_oneregion$daily_income) * 10^(-6)) * 10^6),
                                           by = 10^7)
                                     )
                     )
ggsave(plot = p, filename = paste0("../my_output_files/outliers_billionaires_", world_6region_vec[i], ".pdf"),
       width = 7 * 2)

} else {
set.seed(155)
p <- ggplot() +
  geom_boxplot(data = all_oneregion,
                        mapping = aes(y = daily_income, x = 1),
                        color = "purple", outlier.size = 3)  +
  geom_text_repel(mapping = aes(y = daily_income,
                                x = 1,
                                label = person,
                                size = daily_income),
                  data = outliers_oneregion,
                  max.overlaps = 100,  force = 7,
                  alpha = 0.7,
                  segment.alpha = 0.3,
                  segment.size = 0.2) +
  scale_x_continuous(breaks = NULL,
                     name = world_6region_vec[i]) +
  scale_size_continuous(range = c(6,10),
                        breaks = seq(from = round(min(outliers_oneregion$daily_income)),
                                     to = round(max(outliers_oneregion$daily_income)),
                                     by = round(max(outliers_oneregion$daily_income) * 0.4))) +
  scale_y_continuous(breaks = seq(from = 10^6,
                                  to = ceiling(round(
                                    max(all_oneregion$daily_income) * 10^(-6)) * 10^6),
                                  by = 10^7),
                      labels = as.character(seq(from = 10^6,
                                                to = ceiling(round(
```

```
                                                    max(all_oneregion$daily_income) * 10^(-6)) * 10^6),
                                                  by = 10^7))) +
      geom_hline(yintercept = outliers_above[i],
                 color = "seagreen",
                 linewidth = 1,
                 linetype=3,
                 alpha = 0.4) +
      annotate(geom = "text",
               x = 0.6,
               y = outliers_above[i] + y_axis_offset_for_outlier_label,
               label = outliers_above[i],
               color = "seagreen",
               size = 6) +
      theme(axis.text = element_text(size = 12),
            axis.title = element_text(size = 14))
    cat(world_6region_vec[i], "\n")
    print(p)
    ggsave(plot = p, filename = paste0(
      "../my_output_files/outliers_billionaires_",
      world_6region_vec[i], ".pdf"),
           width = 7 * 2.2)

    }
  }
```

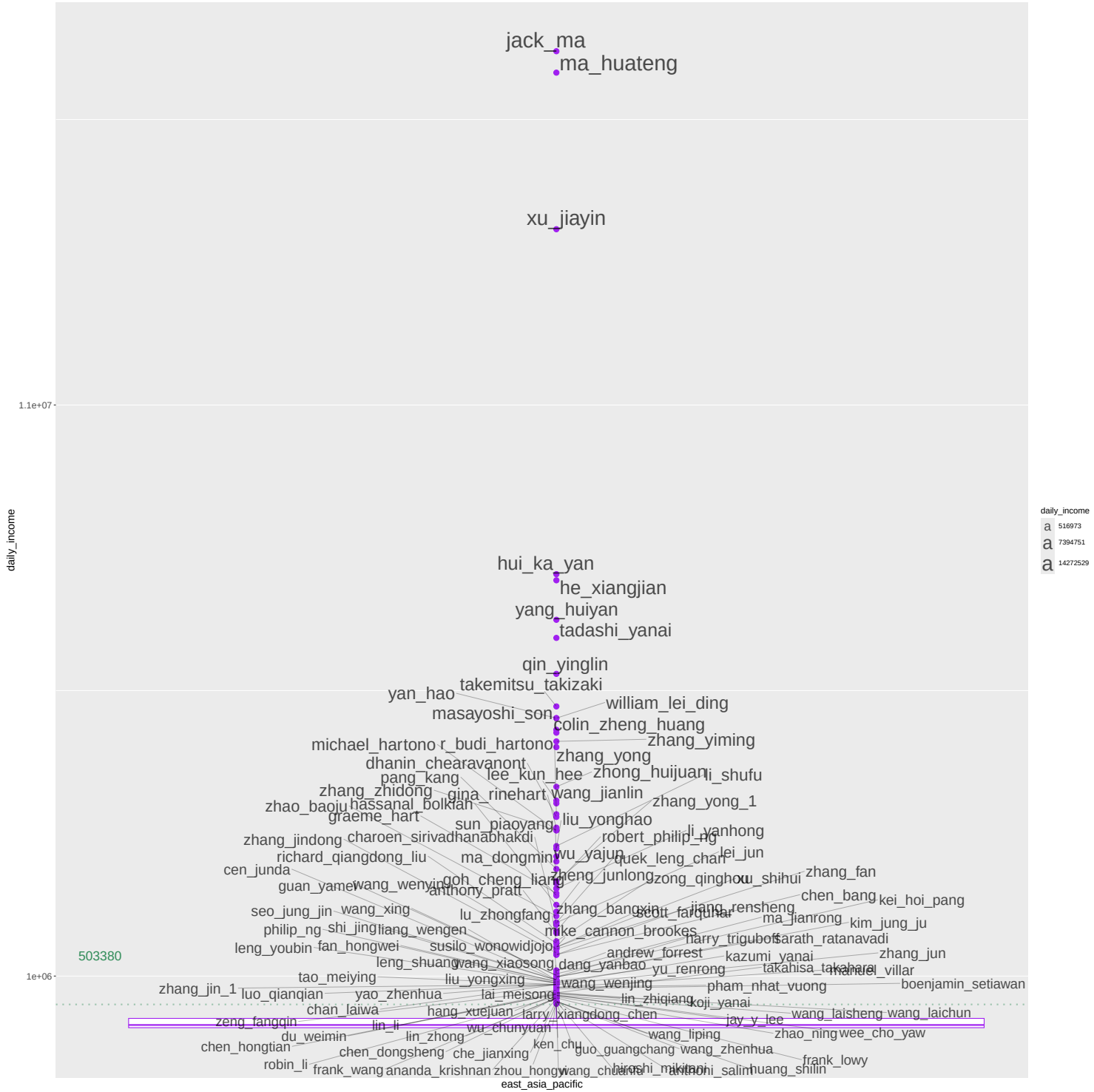
```
america
```

```
Saving 15.4 x 20 in image
```

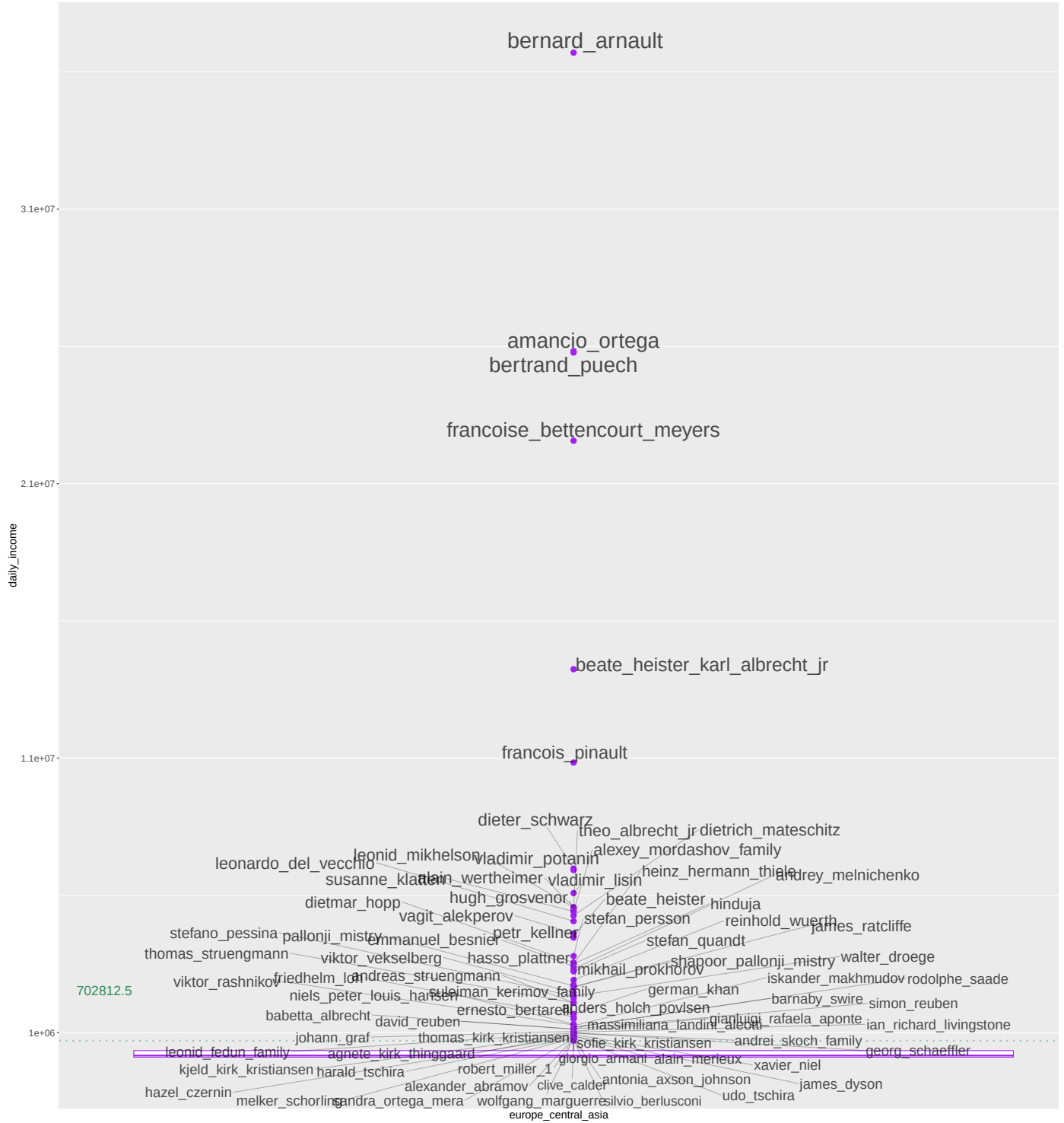

jeff_bezos
bill_gates
warren_buffett
larry_ellison
mark_zuckerberg
rob_walton
jim_walton
steve_ballmer
alice_walton
carlos_slim_helu
larry_page
sergey_brin
michael_bloomberg
charles_koch
julia_koch
mackenzie_scott
david_thomson
phil_knight
sheldon_adelson
john_mars
jacqueline_mars
elon_musk
michael_dell
laurene_powell_jobs
steve_cohen
joseph_safra
jim_simons
ray_dalio
rupert_murdoch
lukas_walton
stephen_schwarzmann
thomas_peterffy
donald_bren
leonard_lauder
carl_cook
jan_koum
richard_b_cohen
ken_griffin
ann_walton_kroenke
david_geffen
dustin_moskovitz
david_duffield
thomas_frist_jr
jude_reyes
jim_kennedy
alberto_bailleres_gonzalez
marcel_herrmann_telles
israel_englander
charles_schwab
victoria_mars
marc_benioff
james_goodnight
pamela_mars
tom_judy_love
paolo_rocca
steven_rales
andrew
herbert_kohler
david_siegel
john_overdeck
valerie_mars
doug_meijer
john_malone
david_green
david_shaw
robert_kraft
eli_broad
leon_black
patrick_soon_shiong
daniel_gilbert
j_christopher_reyes
galen_weston
diane_hendricks
robert_pera
john_doerr
shahid_khan
andrew_beal
stephen_ross
jerry_jones
luis_carlos_sarmiento
stanley_kroenke
george_soros
stewart_and_lynda_resnick
gordon_moore
philip_anschutz
tom_love
emilio_alves_odebrecht
christy_walton
german_larrea_mota_velasco
eric_schmidt
abigail_johnson
donald_newhouse
ricardo_salinas_pliego
sara_mota_de_larrea
hank_doug_meijer
pierre_omidyar
david_tepper
john_menard_jr
carl_icahn
835517
5.1e+07
4.1e+07
3.1e+07
2.1e+07
1.1e+07
1e+06
daily_income
america
daily_income
a 852401
a 23121298
a 45390195

```
east_asia_pacific
```

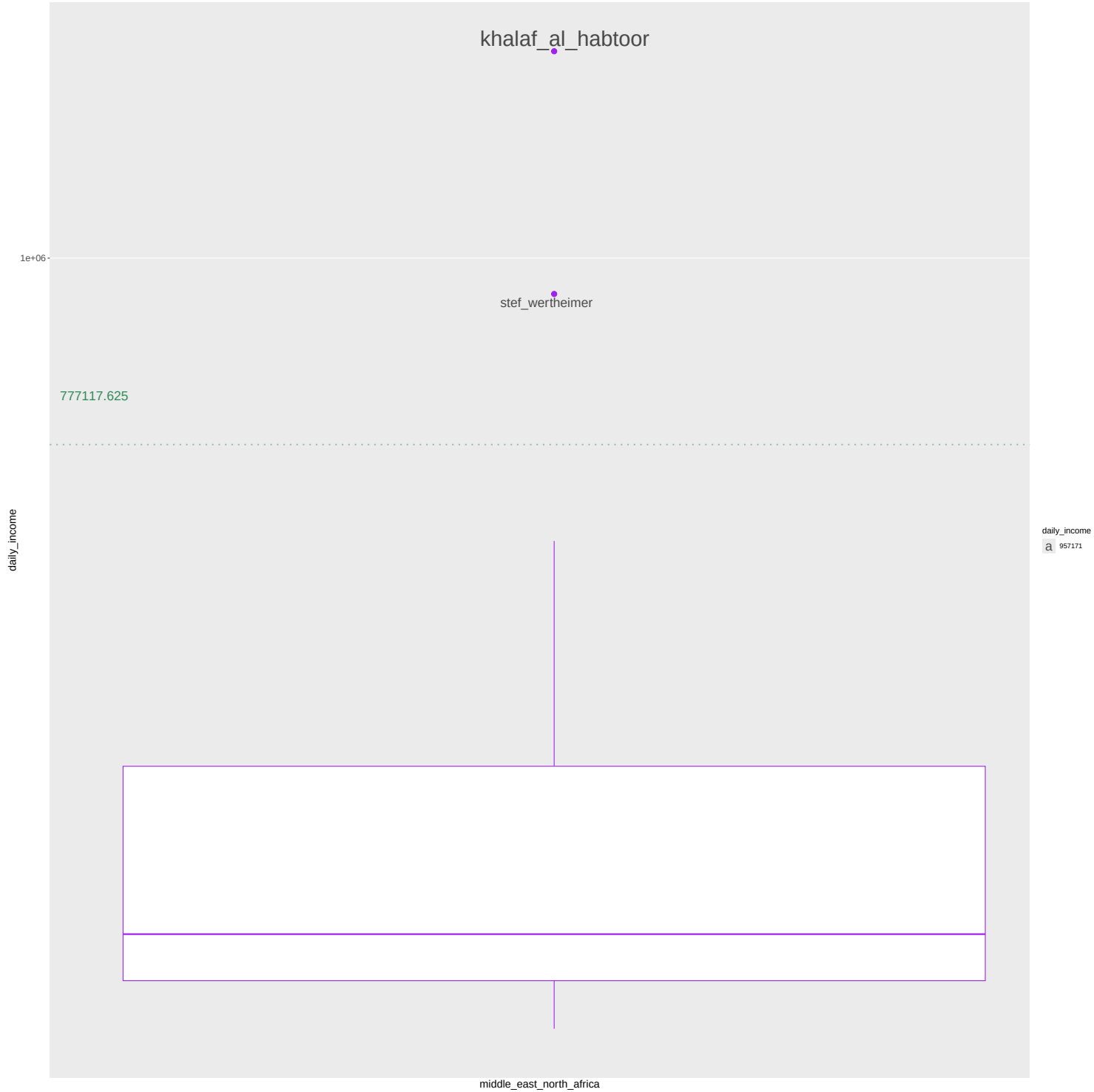
```
Saving 15.4 x 20 in image
```


jack_ma
ma_huateng
xu_jiayin
1.1e+07
daily_income
hui_ka_yan
he_xiangjian
yang_huiyan
tadashi_yanai
qin_yinglin
takemitsu_takizaki
yan_hao
william_lei_ding
masayoshi_son
colin_zheng_huang
michael_hartono
r_budi_hartono
zhang_yiming
zhang_yong
dhanin_chearavanont
lee_kun_hee
zhong_huijuan
li_shufu
pang_kang
zhang_zhidong
gina_rinehart
wang_jianlin
zhao_baoju
hassanal_bolkiah
zhang_yong_1
graeme_hart
sun_piaoyang
liu_yonghao
li_yanhong
zhang_jindong
charoen_sirivadhanabhakdi
robert_philip_ng
richard_qiangdong_liu
ma_dongmin
wu_yajun
quek_leng_chan
lei_jun
cen_junda
zheng_junlong
zong_qinghou
xu_shihui
zhang_fan
guan_yamei
wang_wenying
anthony_pratt
chen_bang
kei_hoi_pang
seo_jung_jin
wang_xing
lu_zhongfang
zhang_bangxin
scott_farquhar
jiang_rensheng
ma_jianrong
kim_jung_ju
philip_ng
shi_jing
liang_wengen
mike_cannon_brookes
harry_triguboff
sarath_ratanavadi
503380
leng_youbin
fan_hongwei
susilo_wonowidjojo
andrew_forrest
kazumi_yanai
zhang_jun
leng_shuang
wang_xiaosong
dang_yanbao
yu_renrong
takahisa_takahara
manuel_villar
tao_meiying
liu_yongxing
1e+06
zhang_jin_1
luo_qianqian
yao_zhenhua
lai_meisong
wang_wenjing
pham_nhat_vuong
boenjamin_setiawan
lin_zhiqiang
koji_yanai
chan_laiwa
hang_xuejuan
larry_xiangdong_chen
wang_laisheng
wang_laichun
zeng_fangqin
lin_li
wu_chunyuan
jay_y_lee
du_weimin
lin_zhong
wang_liping
zhao_ning
wee_cho_yaw
chen_hongtian
ken_chu
guo_guangchang
wang_zhenhua
chen_dongsheng
che_jianxing
frank_lowy
robin_li
frank_wang
ananda_krishnan
zhou_hongyi
wang_chuanfu
hiroshi_mikitani
anthoni_salim
huang_shilin
east_asia_pacific
daily_income
a 516973
a 7394751
a 14272529

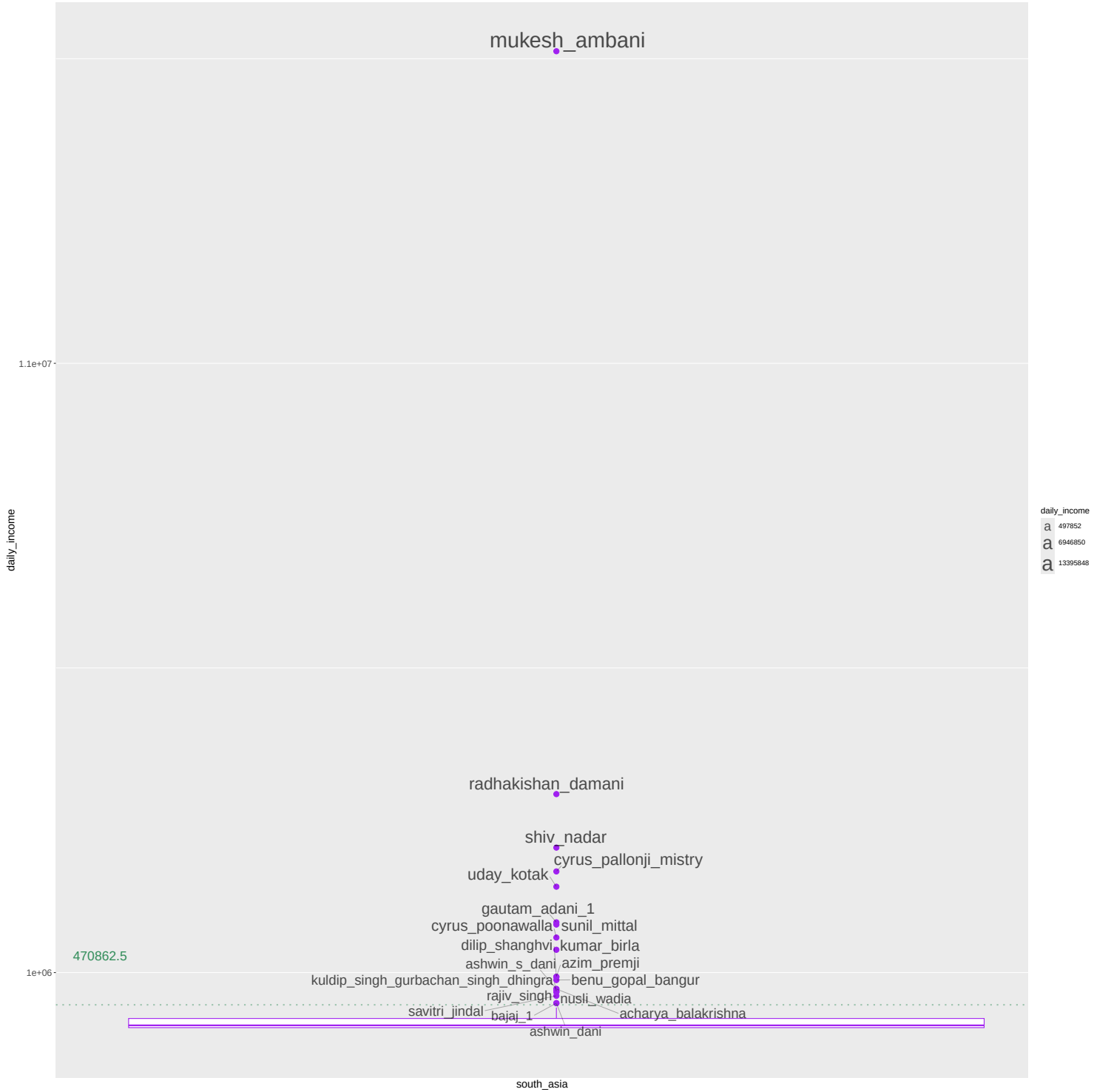
```
europe_central_asia

Saving 15.4 x 20 in image
```


bernard_arnault
amancio_ortega
bertrand_puech
francoise_bettencourt_meyers
beate_heister_karl_albrecht_jr
francois_pinault
3.1e+07
2.1e+07
1.1e+07
1e+06
daily_income
europe_central_asia
702812.5
daily_income
a 706931
a 15387488
a 30068045
dieter_schwarz
theo_albrecht_jr
dietrich_mateschitz
alexey_mordashov_family
leonid_mikhelson
vladimir_potanin
leonardo_del_vecchio
heinz_hermann_thiele
susanne_klatten
alain_wertheimer
vladimir_lisin
andrey_melnichenko
hugh_grosvenor
beate_heister
dietmar_hopp
hinduja
vagit_alekperov
stefan_persson
reinhold_wuerth
james_ratcliffe
stefano_pessina
pallonji_mistry
petr_kellner
emmanuel_besnier
stefan_quandt
thomas_struengmann
viktor_vekselberg
hasso_plattner
shapoor_pallonji_mistry
walter_droege
mikhail_prokhorov
viktor_rashnikov
friedhelm_loh
andreas_struengmann
iskander_makhmudov
rodolphe_saade
niels_peter_louis_hansen
suleiman_kerimov_family
german_khan
barnaby_swire
simon_reuben
babetta_albrecht
ernesto_bertarelli
anders_holch_povlsen
david_reuben
massimiliana_landini_aleotti
gianluigi_rafaela_aponte
ian_richard_livingstone
johann_graf
thomas_kirk_kristiansen
sofie_kirk_kristiansen
andrei_skoch_family
leonid_fedun_family
agnete_kirk_thinggaard
georg_schaeffler
giorgio_armani
alain_merieux
kjeld_kirk_kristiansen
harald_tschira
robert_miller_1
xavier_niel
antonia_axson_johnson
hazel_czernin
alexander_abramov
clive_calder
james_dyson
melker_schorling
sandra_ortega_mera
wolfgang_marguerre
silvio_berlusconi
udo_tschira

```
middle_east_north_africa

Saving 15.4 x 20 in image
```


khalaf_al_habtoor
1e+06
stef_wertheimer
777117.625
daily_income
middle_east_north_africa
daily_income
a 957171

```
south_asia

Saving 15.4 x 20 in image

sub_saharan_africa has no outliers. I will plot all names.

Saving 14 x 20 in image
```


mukesh_ambani
1.1e+07
daily_income
daily_income
a 497852
a 6946850
a 13395848
radhakishan_damani
shiv_nadar
cyrus_pallonji_mistry
uday_kotak
gautam_adani_1
cyrus_poonawalla
sunil_mittal
dilip_shanghvi
kumar_birla
470862.5
ashwin_s_dani
azim_premji
1e+06
kuldip_singh_gurbachan_singh_dhingra
benu_gopal_bangur
rajiv_singh
nusli_wadia
savitri_jindal
bajaj_1
acharya_balakrishna
ashwin_dani
south_asia