Time Series Analysis & Forecasting Using R

Outline

Outline

Strength of seasonality and trend

STL decomposition

$$y_t = T_t + S_t + R_t$$

Seasonal strength

$$\max\left(0, 1 - \frac{\mathsf{Var}(R_t)}{\mathsf{Var}(S_t + R_t)}\right)$$

Trend strength

$$\max\left(0, 1 - \frac{\mathsf{Var}(R_t)}{\mathsf{Var}(T_t + R_t)}\right)$$

Feature extraction and statistics

```
tourism ▷ features(Trips, feat_stl)
```

```
## # A tibble: 304 x 12
##    Region     State Purpose trend_strength seasonal_streng~ seasonal_peak_y~
##    <chr>      <chr> <chr>            <dbl>            <dbl>            <dbl>
##  1 Adelaide   Sout~ Busine~          0.464            0.407                3
##  2 Adelaide   Sout~ Holiday          0.554            0.619                1
##  3 Adelaide   Sout~ Other            0.746            0.202                2
##  4 Adelaide   Sout~ Visiti~          0.435            0.452                1
##  5 Adelaide~  Sout~ Busine~          0.464            0.179                3
##  6 Adelaide~  Sout~ Holiday          0.528            0.296                2
##  7 Adelaide~  Sout~ Other            0.593            0.404                2
##  8 Adelaide~  Sout~ Visiti~          0.488            0.254                0
##  9 Alice Sp~  Nort~ Busine~          0.534            0.251                0
## 10 Alice Sp~  Nort~ Holiday          0.381            0.832                3
## # ... with 294 more rows, and 6 more variables:
## #   seasonal_trough_year <dbl>, spikiness <dbl>, linearity <dbl>,
## #   curvature <dbl>, stl_e_acf1 <dbl>, stl_e_acf10 <dbl>
```
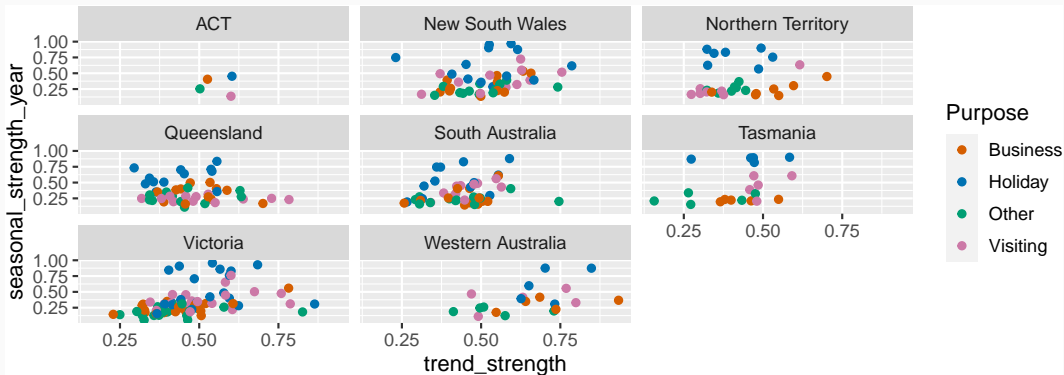
Feature extraction and statistics

```
tourism ▷
  features(Trips, feat_stl) ▷
  ggplot(aes(x = trend_strength, y = seasonal_strength_year, col = Purpose)) +
  geom_point() + facet_wrap(vars(State))
```
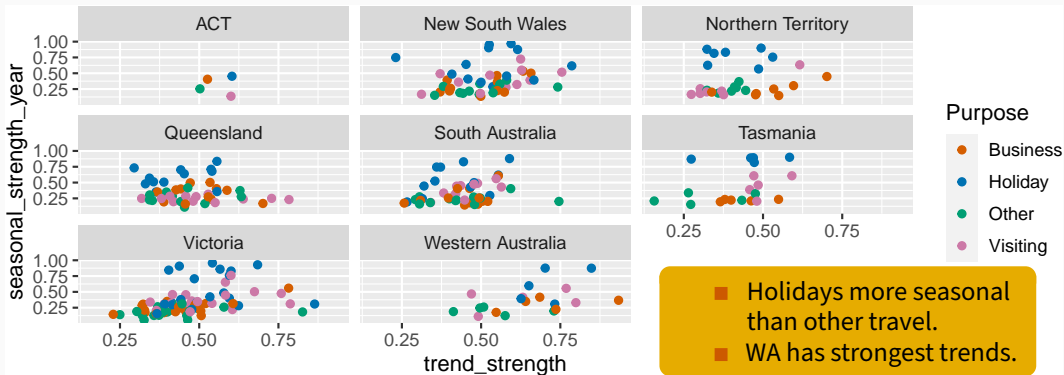
Feature extraction and statistics

```
tourism ▷
  features(Trips, feat_stl) ▷
  ggplot(aes(x = trend_strength, y = seasonal_strength_year, col = Purpose)) +
  geom_point() + facet_wrap(vars(State))
```


- Holidays more seasonal than other travel.
- WA has strongest trends.

Feature extraction and statistics

Find the most seasonal time series:

```r
most_seasonal <- tourism ▷
  features(Trips, feat_stl) ▷
  filter(seasonal_strength_year == max(seasonal_strength_year))
```

Feature extraction and statistics

Find the most seasonal time series:

```
most_seasonal <- tourism ▷
  features(Trips, feat_stl) ▷
  filter(seasonal_strength_year == max(seasonal_strength_year))
```

```
tourism ▷
  right_join(most_seasonal, by = c("State", "Region", "Purpose")) ▷
  ggplot(aes(x = Quarter, y = Trips)) +
  geom_line() + facet_grid(vars(State, Region, Purpose))
```
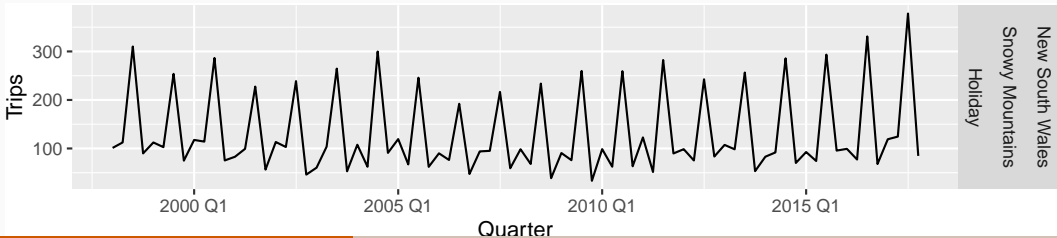
Feature extraction and statistics

Find the most trended time series:

```
most_trended <- tourism |>
  features(Trips, feat_stl) |>
  filter(trend_strength == max(trend_strength))
```

Feature extraction and statistics

Find the most trended time series:

```
most_trended <- tourism ▷
  features(Trips, feat_stl) ▷
  filter(trend_strength == max(trend_strength))
```

```
tourism ▷
  right_join(most_trended, by = c("State", "Region", "Purpose")) ▷
  ggplot(aes(x = Quarter, y = Trips)) +
  geom_line() + facet_grid(vars(State, Region, Purpose))
```
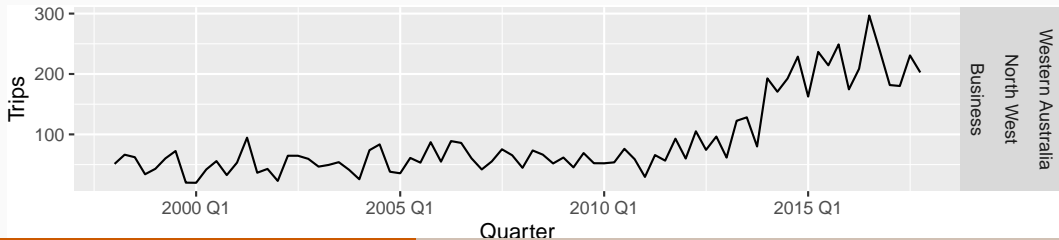
Outline

- Use `GGally::ggpairs()` to look at the relationships between the STL-based features. You might wish to change `seasonal_peak_year` and `seasonal_trough_year` to factors.
- Which is the peak quarter for holidays in each state?

Feature extraction and statistics

```
tourism ▷ features(Trips, feat_acf)
```

```
## # A tibble: 304 x 10
##    Region      State Purpose    acf1 acf10 diff1_acf1 diff1_acf10 diff2_acf1
##    <chr>       <chr> <chr>     <dbl> <dbl>      <dbl>       <dbl>      <dbl>
##  1 Adelaide    Sout~ Busine~  0.0333 0.131     -0.520       0.463     -0.676
##  2 Adelaide    Sout~ Holiday  0.0456 0.372     -0.343       0.614     -0.487
##  3 Adelaide    Sout~ Other    0.517  1.15      -0.409       0.383     -0.675
##  4 Adelaide    Sout~ Visiti~  0.0684 0.294     -0.394       0.452     -0.518
##  5 Adelaide~   Sout~ Busine~  0.0709 0.134     -0.580       0.415     -0.750
##  6 Adelaide~   Sout~ Holiday  0.131  0.313     -0.536       0.500     -0.716
##  7 Adelaide~   Sout~ Other    0.261  0.330     -0.253       0.317     -0.457
##  8 Adelaide~   Sout~ Visiti~  0.139  0.117     -0.472       0.239     -0.626
##  9 Alice Sp~   Nort~ Busine~  0.217  0.367     -0.500       0.381     -0.658
## 10 Alice Sp~   Nort~ Holiday -0.00660 2.11     -0.153       2.11      -0.274
## # ... with 294 more rows, and 2 more variables: diff2_acf10 <dbl>,
## #   season_acf1 <dbl>
```

Outline

Feature extraction and statistics

```
tourism_features <- tourism ▷
  features(Trips, feature_set(pkgs = "feasts"))
```

All features from the feasts package

```
## # A tibble: 304 x 51
##    Region       State Purpose trend~1 seaso~2 seaso~3 seaso~4
##    <chr>        <chr> <chr>     <dbl>   <dbl>   <dbl>   <dbl>
##  1 Adelaide     Sout~ Busine~   0.464   0.407       3       1
##  2 Adelaide     Sout~ Holiday   0.554   0.619       1       2
##  3 Adelaide     Sout~ Other     0.746   0.202       2       1
##  4 Adelaide     Sout~ Visiti~   0.435   0.452       1       3
##  5 Adelaide H~  Sout~ Busine~   0.464   0.179       3       0
##  6 Adelaide H~  Sout~ Holiday   0.528   0.296       2       1
##  7 Adelaide H~  Sout~ Other     0.593   0.404       2       2
##  8 Adelaide H~  Sout~ Visiti~   0.488   0.254       0       3
##  9 Alice Spri~  Nort~ Busine~   0.534   0.251       0       1
## 10 Alice Spri~  Nort~ Holiday   0.381   0.832       3       1
## # ... with 294 more rows, 44 more variables:
## #   spikiness <dbl>, linearity <dbl>, curvature <dbl>,
## #   stl_e_acf1 <dbl>, stl_e_acf10 <dbl>, acf1 <dbl>,
```

13

Feature extraction and statistics

```
pcs <- tourism_features ▷
  select(-State, -Region, -Purpose) ▷
  prcomp(scale = TRUE) ▷
  broom::augment(tourism_features)
```
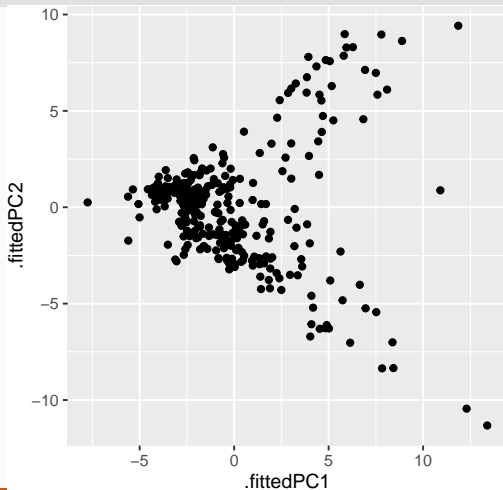
Principal components based on all features from the feasts package

```
## # A tibble: 304 x 100
##    .rownames Region     State Purpose trend~1 seaso~2 seaso~3
##    <chr>     <chr>      <chr> <chr>     <dbl>   <dbl>   <dbl>
## 1  1         Adelaide   Sout~ Busine~   0.464   0.407       3
## 2  2         Adelaide   Sout~ Holiday   0.554   0.619       1
## 3  3         Adelaide   Sout~ Other     0.746   0.202       2
## 4  4         Adelaide   Sout~ Visiti~   0.435   0.452       1
## 5  5         Adelaide~  Sout~ Busine~   0.464   0.179       3
## 6  6         Adelaide~  Sout~ Holiday   0.528   0.296       2
## 7  7         Adelaide~  Sout~ Other     0.593   0.404       2
## 8  8         Adelaide~  Sout~ Visiti~   0.488   0.254       0
## 9  9         Alice Sp~  Nort~ Busine~   0.534   0.251       0
## 10 10        Alice Sp~  Nort~ Holiday   0.381   0.832       3
## # ... with 294 more rows, 93 more variables:
```

14

Feature extraction and statistics

```
pcs ▷ ggplot(aes(x=.fittedPC1, y=.fittedPC2)) +
  geom_point() + theme(aspect.ratio=1)
```
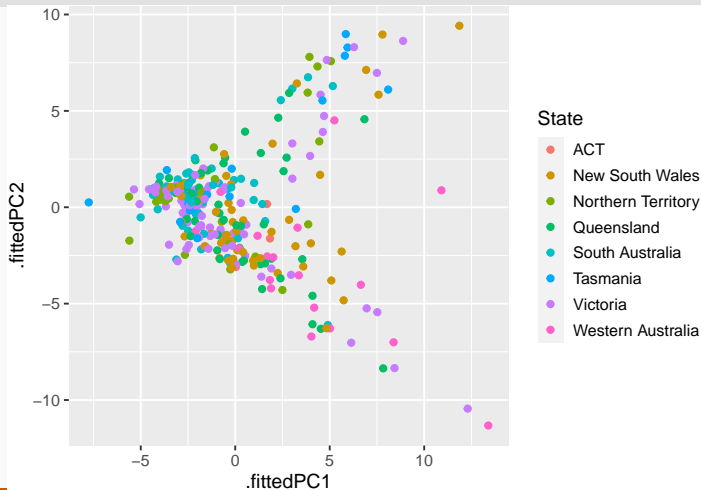
Principal components based on all features from the feasts package

Feature extraction and statistics

```
pcs ▷ ggplot(aes(x=.fittedPC1, y=.fittedPC2, col=State)) +
  geom_point() + theme(aspect.ratio=1)
```
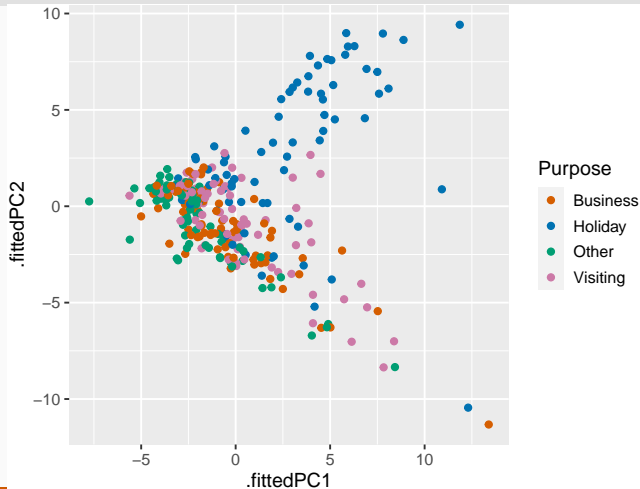
Principal components based on all features from the feasts package

Feature extraction and statistics

```
pcs ▷ ggplot(aes(x=.fittedPC1, y=.fittedPC2, col=Purpose)) +
  geom_point() + theme(aspect.ratio=1)
```
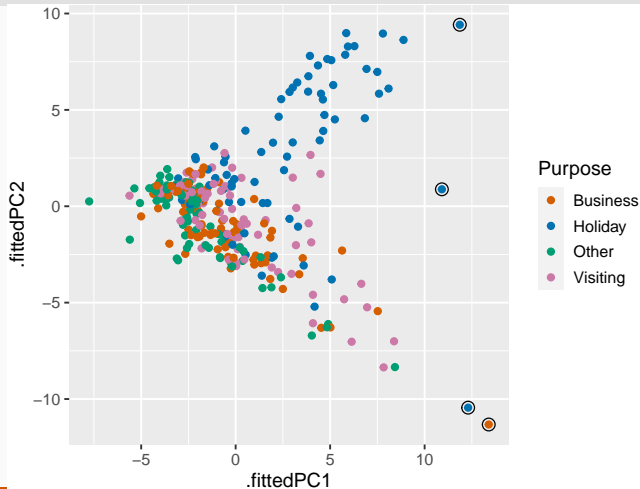
Principal components based on all features from the feasts package

Feature extraction and statistics

```
pcs ▷ ggplot(aes(x=.fittedPC1, y=.fittedPC2, col=Purpose)) +
  geom_point() + theme(aspect.ratio=1)
```
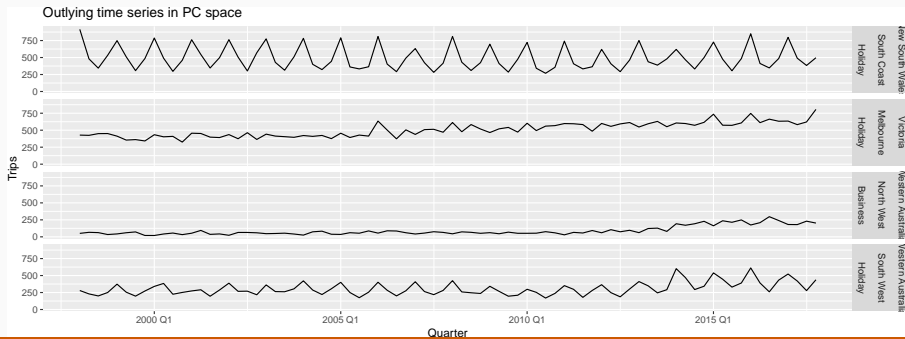
Principal components based on all features from the feasts package

Feature extraction and statistics

```
outliers ▷
  left_join(tourism, by = c("State", "Region", "Purpose")) ▷
  mutate(Series = glue("{State}", "{Region}", "{Purpose}", .sep = "\n\n")) ▷
  ggplot(aes(x = Quarter, y = Trips)) +
  geom_line() + facet_grid(Series ~ .) +
  labs(title = "Outlying time series in PC space")
```

Outline

- Use a feature-based approach to look for outlying series in PBS.
- What is unusual about the series you identify as outliers?