Stat 443. Time Series and Forecasting.

Key ideas: Seasonal exponential smoothing.

Three recursion equations: one for level, one for slope, one for seasonal. The seasonal effect could be additive or multiplicative.

These slides mainly cover the additive case.

The technique used to get a stochastic model for the additive case does not work for the multiplicative case.

Winters additive seasonal. The recursions and h-step forecasts are as follows. Let d be the periodicity length. For $t > d$,

$$
\begin{aligned}
\hat{\ell}_t &= \alpha(y_t - \hat{s}_{t-d}) + (1-\alpha)(\hat{\ell}_{t-1} + \hat{b}_{t-1}) \\
\hat{b}_t &= \beta(\hat{\ell}_t - \hat{\ell}_{t-1}) + (1-\beta)\hat{b}_{t-1} \\
\hat{s}_t &= \gamma(y_t - \hat{\ell}_t) + (1-\gamma)\hat{s}_{t-d} \\
\hat{y}_{t+1|t} &= \hat{s}_{t+1-d} + (\hat{\ell}_t + \hat{b}_t), \\
\hat{y}_{t+2|t} &= \hat{s}_{t+2-d} + (\hat{\ell}_t + 2\hat{b}_t), \\
\hat{y}_{t+h|t} &= \hat{s}_{t+h-d} + (\hat{\ell}_t + h\hat{b}_t), \quad h = 1, \ldots, d \\
\hat{y}_{t+d+1|t} &= \hat{s}_{t+1-d} + (\hat{\ell}_t + (d+1)\hat{b}_t), \text{ etc.}
\end{aligned}
$$

The deseasonalized smoothed level $\hat{\ell}_t$ is a convex combination of the *most recent deseasonalized observation* and the *local linear projection*.
The smoothed slope $\hat{b}_t$ is a convex combination of the *most recent slope change* and the *previous smoothed slope*.
The smoothed seasonal effect $\hat{s}_t$ is a convex combination of the *most recent seasonal estimate* and the *previous seasonal effect*.
The h-step forecast is based on the most recent seasonal effect (at the same position in the seasonal cycle), plus a constant slope and the last level value.

How are recursions initialized?

For all exponential smoothing methods, check documentation on how the smoothed series are initialized. For example, how are $\widehat{\ell}_1, \widehat{b}_1$ determined.

Winter multiplicative seasonal: see recursion equations in the document stat443-expsmo.pdf in the Reference Notes section at the course web site.

stat443-expsmo.pdf has a summary of recursions for all of the exponential smoothing methods; also it has the input parameters for `HoltWinters` in R and `PROC ESM` in SAS.

Comments for Vancouver precipitation example

Note: total precipitation at Vancouver airport in mm = rainfall + snowfall in water equivalent after melting.

This series is more irregular than the series for Vancouver average monthly temperature; for the latter, Winters additive seasonal forecast rule does better than simple rules.
For total monthly precipitation, there is much more variation in some months than others (especially for months with more precipitation).
Forecasts can underestimate the maxima by a lot in winter months.
With non-constant variance for noise variables or innovations, rmse may not such a good measure of forecast accuracy.

Box-Jenkins forecasting methods assume constant variance for noise variables or innovations.

For the precipitation data, one could compare forecasting rules with square root precipitation.

Example 1: sqrt(total precipitation by month) over 86 years, Vancouver YVR

month	1	2	3	4	5	6	7	8	9	10	11	12
SD(precip)	57.7	48.4	42.9	34.6	28.9	27.5	21.9	29.0	39.4	56.0	66.3	50.3
SD(sqrt(precip))	2.51	2.31	2.13	2.10	2.02	2.04	2.24	2.37	2.67	2.60	2.62	2.01

```
Output with HoltWinters() for training set.
additive                multiplicative
 alpha: 0.0621           alpha: 0.0498          little change in level?
 beta : 0.0145           beta : 0.0208          little change in slope?
 gamma: 0.1044           gamma: 0.1706
Coefficients:           Coefficients:
a    10.0435            a    12.1184            last level in training set
b     0.0026            b     0.0069            last slope in training set
s1    2.9205            s1    1.1112            seasonal peak multiplicative
s2   -0.7494            s2    0.7233
s3    0.5382            s3    0.8931
s4   -1.1677            s4    0.7244
s5   -1.9920            s5    0.6695
s6   -3.2211            s6    0.5403
s7   -4.0294            s7    0.4904
s8   -4.4398            s8    0.4535             seasonal trough
s9   -2.9110            s9    0.6158
s10   1.0619            s10   0.9491
s11   2.9167            s11   1.0716
s12   3.0247            s12   1.0867             seasonal peak additive
```

Forecasting rules for Vancouver monthly total precipitation 1938–2005 as training set, 2006–2023 as holdout.

yearmon	holdout	addseasonal	multseasonal	persistmon	iidbymon
2006.01	283.6	171.16	190.35	249.6	151.10
2006.02	57.0	96.54	74.10	45.8	114.73
2006.03	92.4	117.55	120.01	132.8	103.88
2006.04	70.0	86.37	81.60	90.2	72.55
2006.05	42.8	67.83	65.61	68.6	57.44
2006.06	54.4	49.38	41.56	49.6	47.67
2006.07	25.2	39.88	36.11	43.6	33.96
2006.08	4.8	37.68	37.15	28.6	37.37
2006.09	39.4	56.03	69.28	53.6	57.45
2006.10	57.8	128.59	143.20	155.4	119.04
2006.11	350.8	162.26	161.34	136.6	158.98
2006.12	146.0	175.60	169.24	160.8	172.26
...					
2023.12					
rmse		45.3	47.6	64.1	44.1

Forecasting rules for Vancouver monthly sqrt(total precipitation)
1938–2005 as training set, 2006–2023 as holdout.

yearmon	holdout	addseasonal	multseasonal	persistmon	iidbymon
2006.01	16.84	12.97	13.47	15.80	12.01
2006.02	7.55	9.54	8.89	6.77	10.43
2006.03	9.61	10.71	10.90	11.52	10.01
2006.04	8.37	8.94	8.79	9.50	8.25
2006.05	6.54	8.08	8.11	8.28	7.34
2006.06	7.38	6.76	6.48	7.04	6.57
2006.07	5.02	5.99	5.93	6.60	5.44
2006.08	2.19	5.52	5.44	5.35	5.65
2006.09	6.28	6.84	7.17	7.32	7.14
2006.10	7.60	10.78	10.97	12.47	10.58
2006.11	18.73	12.43	12.19	11.69	12.34
2006.12	12.08	12.93	12.69	12.68	12.98
...					
2023.12					
rmse		2.37	2.42	3.32	2.31

Example 2: monthly mean temperature over 86 years, Vancouver YVR

month	1	2	3	4	5	6	7	8	9	10	11	12
SD(precip)	2.21	1.50	1.16	0.94	1.08	1.04	1.02	0.98	0.97	0.91	1.47	1.63

```
Output with HoltWinters() for training set.
additive                multiplicative
alpha: 0.1045            alpha: 0.9583            larger change in mult level
beta : 0                 beta : 0                 little change in slope
gamma: 0.1150            gamma: 0.5881            larger change in mult seasonal
Coefficients:           Coefficients:
a    10.8576            a    21.2854              last level in training set
b     0.0011            b     0.0011              last slope in training set
s1   -6.3633            s1    0.2499              seasonal trough additive
s2   -5.6399            s2    0.3125
s3   -3.5317            s3    0.4816
s4   -0.8825            s4    0.7383
s5    2.4039            s5    1.1282
s6    5.3124            s6    1.4788
s7    7.5527            s7    1.7278
s8    7.6558            s8    1.7306              seasonal peak
s9    4.4260            s9    1.4445
s10  -0.0416            s10   1.0284
s11  -4.1258            s11   0.3962
s12  -6.3402            s12   0.2148              seasonal trough multiplicative
```

Forecasting rules for Vancouver monthly average temperature 1938–2005 as training set, 2006–2023 as holdout.

yearmon	holdout	addseasonal	multseasonal	persistmon	iidbymon
2006.01	6.30	4.50	5.32	3.71	3.06
2006.02	4.28	5.41	7.83	4.30	4.58
2006.03	6.55	7.40	6.83	8.36	6.30
2006.04	9.30	9.96	10.06	10.12	9.13
2006.05	13.04	13.18	14.26	14.34	12.49
2006.06	16.67	16.07	17.16	15.60	15.31
2006.07	18.68	18.38	19.50	18.12	17.52
2006.08	17.56	18.51	18.75	18.97	17.42
2006.09	15.26	15.19	14.70	14.67	14.48
2006.10	10.04	10.73	10.85	11.29	10.14
2006.11	5.74	6.57	3.88	5.65	6.09
2006.12	4.47	4.27	3.07	4.60	3.83
...					
2023.12					
rmse		1.22	1.84	1.63	1.34

Pseudo-code for rmse (Winters additive seasonal exponential smoothing)

- Input `train` with size n, periodicity length d; n multiple of d.
- Estimate α, β, γ parameters as $\widehat{\alpha}$, $\widehat{\beta}$. $\widehat{\gamma}$, and get the 3 smoothed series $\{\widehat{\ell}_t\}$; $\{\widehat{b}_t\}$; $\{\widehat{s}_t\}$. $\widehat{\ell}_n$ is the last smoothed level value of the training set. $\widehat{b}_n$ is the last smoothed slope value of the training set. $\widehat{s}_{n-d+1}, \ldots, \widehat{s}_n$ are the smoothed seasonal values in the last cycle (these are in the `$coefficents` component of `HoltWinters` output).
- Save $\widehat{\alpha}$, $\widehat{\beta}$, $\widehat{\gamma}$, $\widehat{\ell}_n$, $\widehat{b}_n$, $\widehat{s}_{n-d+1}, \ldots, \widehat{s}_n$.

Separate out-of-sample rmse from additive seasonal exponential smoothing

- Input $\widehat{\alpha}$, $\widehat{\beta}$, $\widehat{\gamma}$, $\widehat{\ell}_n$, $\widehat{b}_n$, $s_1 = \widehat{s}_{n-d+1}, \ldots, s_d = \widehat{s}_n$; `holdout` with size $n_{holdout}$ (multiple of d).
- sse$\leftarrow$ 0
- fc$\leftarrow$ $\widehat{\ell}_n + \widehat{b}_n + s_1$; yt$\leftarrow$ holdout[1]; newfc$\leftarrow$ fc; fcvec[1]$\leftarrow$ fc; fcerror$\leftarrow$ yt-fc; sse$\leftarrow$ sse$+$ fcerror^2; ellprev $\leftarrow \widehat{\ell}_n$; bprev $\leftarrow \widehat{b}_n$.
- for i in $2, \ldots, n_{holdout}$:
- $\ell_{new} \leftarrow \widehat{\alpha}\times$ (holdout[i-1]$-s_{(i-1) \bmod d}$) $+(1-\widehat{\alpha}) \times (\ell_{prev} + b_{prev})$; $b_{new} \leftarrow \widehat{\beta}(\ell_{new} - \ell_{prev}) + (1-\widehat{\beta})b_{prev}$; $s_{new} \leftarrow \widehat{\gamma}$(holdout[i-1]-$\ell_{new}$)$+(1-\widehat{\gamma})s_{(i-1) \bmod d}$; fc$\leftarrow$ $\ell_{new} + b_{new} + s_{i \bmod d}$; fcvec[i]$\leftarrow$ fc; yt$\leftarrow$ holdout[i]; fcerror$\leftarrow$ yt-fc; sse$\leftarrow$ sse$+$ fcerror^2;
- $\ell_{prev} \leftarrow \ell_{new}$; $b_{prev} \leftarrow b_{new}$; $s_{(i-1) \bmod d} \leftarrow s_{new}$.
- end for; return rmse=sqrt(sse/$n_{holdout}$)

Note: if $x \bmod d = 0$ replaced by d.

Stochastic model for additive seasonal exponential smoothing

Next, convert additive seasonal recursion equations into a stochastic model. Then after differencing, the model becomes a linear combination of previous innovations up to the previous $(1+d)$th, where d is the periodicity length (e.g., $d = 12$ for monthly data).

(upper case Y, L, B, S and ϵ for random variables)

Stochastic model for additive seasonal exponential smoothing

Write a stochastic model as:

$$
\begin{aligned}
L_t &= \alpha(Y_t - S_{t-d}) + (1-\alpha)(L_{t-1} + B_{t-1}) = L_{t-1} + B_{t-1} + \alpha(Y_t - S_{t-d} - L_{t-1} - B_{t-1}), \\
B_t &= \beta(L_t - L_{t-1}) + (1-\beta)B_{t-1} = B_{t-1} + \beta(L_t - L_{t-1} - B_{t-1}), \\
S_t &= \gamma(Y_t - L_t) + (1-\gamma)S_{t-d} = S_{t-d} + \gamma(Y_t - L_t - S_{t-d}), \\
Y_t &= S_{t-d} + L_{t-1} + B_{t-1} + \epsilon_t \\
Y_{t+1} &= S_{t+1-d} + L_t + B_t + \epsilon_{t+1}
\end{aligned}
$$

Hence

$$
\begin{aligned}
\Delta Y_{t+1} &= Y_{t+1} - Y_t = S_{t+1-d} - S_{t-d} + L_t - L_{t-1} + B_t - B_{t-1} + \epsilon_{t+1} - \epsilon_t \\
&= S_{t+1-d} - S_{t-d} + [B_{t-1} + \alpha(Y_t - S_{t-d} - L_{t-1} - B_{t-1})] + \beta(L_t - L_{t-1} - B_{t-1}) + \epsilon_{t+1} - \epsilon_t \\
&= S_{t+1-d} - S_{t-d} + [B_{t-1} + \alpha\epsilon_t] + \beta\alpha\epsilon_t + \epsilon_{t+1} - \epsilon_t \\
&= B_{t-1} + S_{t+1-d} - S_{t-d} + \epsilon_{t+1} + (\alpha + \alpha\beta - 1)\epsilon_t \\
\Delta Y_{t+1+d} &= B_{t+d-1} + S_{t+1} - S_t + \epsilon_{t+1+d} + (\alpha + \alpha\beta - 1)\epsilon_{t+d}
\end{aligned}
$$

Take a second difference for period d apart:

$$
\begin{aligned}
\Delta Y_{t+1+d} - \Delta Y_{t+1} &= B_{t+d-1} - B_{t-1} + [S_{t+1-d} - S_{t+1}] - [S_{t-d} - S_t] \\
&\quad + \epsilon_{t+1+d} - \epsilon_{t+1} - (1 - \alpha - \alpha\beta)(\epsilon_{t+d} - \epsilon_t) \\
B_{t+d-1} - B_{t-1} &= \sum_{i=0}^{d-1} (B_{t+i} - B_{t+i-1}) = \beta \sum_{i=0}^{d-1} (L_{t+i} - L_{t+i-1} - B_{t+i-1}) \\
&= \beta\alpha \sum_{i=0}^{d-1} (Y_{t+i} - S_{t+i-d} - L_{t+i-1} - B_{t+i-1}) = \alpha\beta \sum_{i=0}^{d-1} \epsilon_{t+i}
\end{aligned}
$$

$$
\begin{aligned}
L_t &= L_{t-1} + B_{t-1} + \alpha(Y_t - S_{t-d} - L_{t-1} - B_{t-1}), \\
S_t &= \gamma(Y_t - L_t) + (1-\gamma)S_{t-d} = S_{t-d} + \gamma(Y_t - S_{t-d} - L_t), \\
Y_t &= S_{t-d} + L_{t-1} + B_{t-1} + \epsilon_t;
\end{aligned}
$$

$$
\begin{aligned}
&(S_{t+1-d} - S_{t+1}) - [S_{t-d} - S_t] = \gamma(Y_{t+1} - S_{t+1-d} - L_{t+1}) - \gamma(Y_t - S_{t-d} - L_t)] \\
&= \gamma(L_t + B_t + \epsilon_{t+1} - L_{t+1}) - \gamma(L_{t-1} + B_{t-1} + \epsilon_t - L_t) \\
&= \gamma(\epsilon_{t+1} - \epsilon_t) - \gamma\alpha(Y_{t+1} - S_{t+1-d} - L_t - B_t) + \gamma\alpha(Y_t - S_{t-d} - L_{t-1} - B_{t-1}) \\
&= \gamma(\epsilon_{t+1} - \epsilon_t) - \gamma\alpha(\epsilon_{t+1} - \epsilon_t) = \gamma(1-\alpha)(\epsilon_{t+1} - \epsilon_t)
\end{aligned}
$$

Combining all of the terms to get an expression in $\{\epsilon_t\}$:

$$
\begin{aligned}
\Delta Y_{t+1+d} - \Delta Y_{t+1} &= \alpha\beta\Sigma_{i=0}^{d-1}\epsilon_{t+i} + \gamma(1-\alpha)(\epsilon_{t+1} - \epsilon_t) + \epsilon_{t+1+d} - \epsilon_{t+1} \\
&\quad -(1-\alpha-\alpha\beta)(\epsilon_{t+d} - \epsilon_t) \\
&= \epsilon_{t+1+d} - \Sigma_{i=d}^{0}\theta_i\epsilon_{t+i}
\end{aligned}
$$

The coefficients θ of $\epsilon_{t+1+d}, \epsilon_{t+d}, \ldots, \epsilon_{t+1}, \epsilon_t$ in the linear combination are:

- ϵ_{t+1+d}: 1
- ϵ_{t+d}: $(1-\alpha-\alpha\beta)$
- ϵ_{t+j} $(j = 2, \ldots, d-1)$: $-\alpha\beta$
- ϵ_{t+1}: $[1 - \alpha\beta - (1-\alpha)\gamma]$
- ϵ_t: $-(1-\alpha)(1-\gamma)$

Note: multiplicative seasonal exponential smoothing does not have an additive innovation.

h-step ahead prediction intervals at end of training set

See Rmd file YVR-monthlytemp.pdf for more details. A few lines are extracted here.

```
vtrain = v$meantemp[1:ntrain]
z = ts(vtrain,start=c(1938,1),frequency=12)
wafit = HoltWinters(z,seasonal="additive")
wmfit = HoltWinters(z,seasonal="multiplicative"

class(wafit)
[1] "HoltWinters"

# predict method: see help(predict.HoltWinters)
wa_pred = predict(wafit, n.ahead=14, prediction.interval=T, level=0.90)
wm_pred = predict(wmfit, n.ahead=14, prediction.interval=T, level=0.90)

mon_ahead = v$yearmon[(ntrain+1):(ntrain+14)]
pred_df = as.data.frame(cbind(mon_ahead/100,wa_pred,wm_pred))
names(pred_df) = c("yearmon","pt_add","upr_add","lwr_add","pt_mul","upr_mul","lwr_mul")
print(round(pred_df,3)) + SEs
```

```
print(round(pred_df,3)) + SEs
    yearmon pt_add upr_add lwr_add pt_mul upr_mul  lwr_mul    SE_add
 1  2006.01  4.495  6.637  2.353   5.319   8.745    1.893    1.302
 2  2006.02  5.220  7.373  3.066   6.651  12.054    1.248    1.309
 3  2006.03  7.329  9.494  5.164  10.253  19.182    1.324    1.316
 4  2006.04  9.979 12.156  7.803  15.718  29.779    1.657    1.323
 5  2006.05 13.267 15.455 11.079  24.020  45.748    2.292    1.330
 6  2006.06 16.176 18.376 13.977  31.486  60.154    2.818    1.337
 7  2006.07 18.418 20.629 16.207  36.789  70.447    3.131    1.344
 8  2006.08 18.522 20.744 16.300  36.852  70.730    2.973    1.351
 9  2006.09 15.293 17.527 13.060  30.761  59.239    2.284    1.358
 10 2006.10 10.827 13.071  8.582  21.902  42.457    1.347    1.365
 11 2006.11  6.743  8.999  4.488   8.438  17.061   -0.185    1.371
 12 2006.12  4.530  6.797  2.263   4.576 701.514 -692.363    1.378
 13 2007.01  4.508  6.818  2.198   5.322 795.697 -785.053    1.404
```

cv = qnorm(0.95)
SE_add = (pred_df$upr_add - pred_df$lwr_add)/(2*cv)