To give everyone interested a better insight into what I did, you find my model documentation attached.

If you have any questions, please ask them here on the forum. I am currently getting a lot of e-mails that I will not be able to answer before Christmas ;)

Here is the dictionary that contains all of the features I used (that's what it is all about!):

```
features={ 'store':[0,0,1,0],'storetype':[0,0,1,0],'assortment':[1,1,1,0], 'shopavg_open':[0,0,1,0],
'shopavg_salespercustomer':[0,0,1,0],
'shopavg_schoolholiday':[1,1,1,0],
'shopsales_holiday':[0,0,1,0],
'shopsales_promo':[1,1,1,0],
'shopsales_saturday':[0,0,1,0],
'day':[1,1,1,0],'da
yofweek':[1,1,1,0],
'dayofyear':[1,1,1,0],
'month':[0,0,1,0],
'week':[1,1,1,0],
'year':[0,0,1,0],
'dayavg_openyesterday':[0,0,1,0],
'Promo2':[0,0,1,0],
'Promo2SinceWeek':[0,0,1,0],
'Promo2SinceYear':[1,1,1,0],
'daysinpromocycle':[1,1,1,0],
'primpromocycle':[1,1,1,0],
'promo':[0,0,1,0],
'promointerval':[1,1,1,0],
'CompetitionDistance':[1,1,1,0],
'CompetitionOpenSinceMonth':[0,0,1,0],
'CompetitionOpenSinceYear':[1,1,1,0],
'daysincompetition':[0,0,1,0],
'daysincompetition_unrounded':[1,1,1,0],
'rnd_CompetitionDistance':[0,0,1,0],
'schoolholiday':[1,1,1,0],
'stateholiday':[1,1,1,0],
'holidays_lastweek':[1,1,1,0],
'holidays_nextweek':[1,1,1,0],
'holidays_thisweek':[1,1,1,0],
'prevquarter_dps_med':[0,0,0,1],
'prevquarter_ds_hmean':[0,0,0,0],
'prevquarter_hmean':[0,0,0,0],
'prevquarter_med':[1,0,0,1],
'prevhalfyear':[0,0,0,1],
'prevhalfyear_m1':[0,0,0,1],
'prevhalfyear_m3':[0,0,0,0],
'prevyear_dphs_med':[0,0,0,1],
'prevyear_dps_med':[1,0,0,1],
'prevyear_ds_m1':[0,0,0,1],
'prevyear_ds_m2ln':[0,0,0,0],
'prevyear_ds_med':[0,0,0,1],
'prevyear_ds_p10':[0,0,0,1],
'prevyear_m1':[0,0,0,1],
'prevyear_m2':[0,0,0,0],
'prevyear_m3':[0,0,0,0],
```

'prevyear_m4':[0,0,0,0],
'prevyear_med':[0,0,0,1],
'prevquarter_cust_dps_med':[0,1,0,1],
'prevyear_cust_dps_med':[0,1,0,1],
'lastmonth_yoy':[0,0,0,0],
'linmod_quarterly':[0,0,0,1],
'linmod_yearly':[0,0,0,1],
'weather_maxtemp':[1,1,1,0],
'weather_precip':[1,1,1,0],
'relativeweather_maxtemp':[0,0,1,0],
'relativeweather_precip':[0,0,1,0], 'closurefeat':[1,1,1,0], }

Meaning of dict entries:

index 0: feature is part of salesmodel
index 1: feature is part of customermodel
index 2: feature is part of MA (monthahead) models
index 3: pmd (previousmonthdate) variant of feature is part of MA models
There is also an 'allfeatures' model, containing all dict entries; and models on months May to September have an extra feature on days relative to summer holiday start

Explanation about features starting with prev:

ds=by day/store
dps= by day/promo/store
dphs=by day/promo/shoolholiday/store
(none)=by promo/store (how I started)
Second part:

med (or none)=median
m1=mean
m2=standard deviation
m3=skewness
m4=kurtosis
hmean=harmonic mean
cust means that it is calculated on customers instead of sales

About the rnd/rounded features: at some point I excluded store from my model, and was worried that daysincompetition and competitiondistance would be used as proxies for store, therefore I rounded them on the log scale. In my final model I don't think it makes a difference.

Relative weather is weathervariable today, divided by average weathervariable during the last 7 days (did not help much).