FUTURE UPDATE

Electric vehicle adoption

When will EV become mainstream?

Collected data

1. Global Internal Combustion Engine car sales
2. Global Electric Vehicle sales
3. Lithium price forecast
4. Battery raw material evolution
5. Diesel sales evolution
6. Hybrid car sales evolution

Creating our data models

Battery technology + price

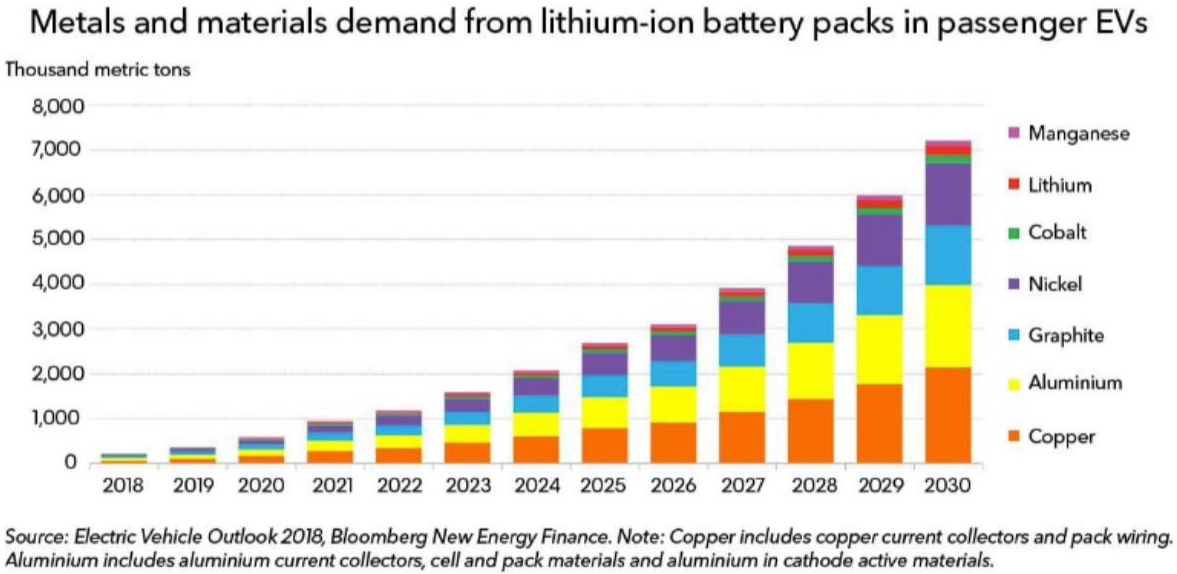
Market development

Substitute technologies

Technologies used

- Python + Jupyter
- R studio
- Scikit learn

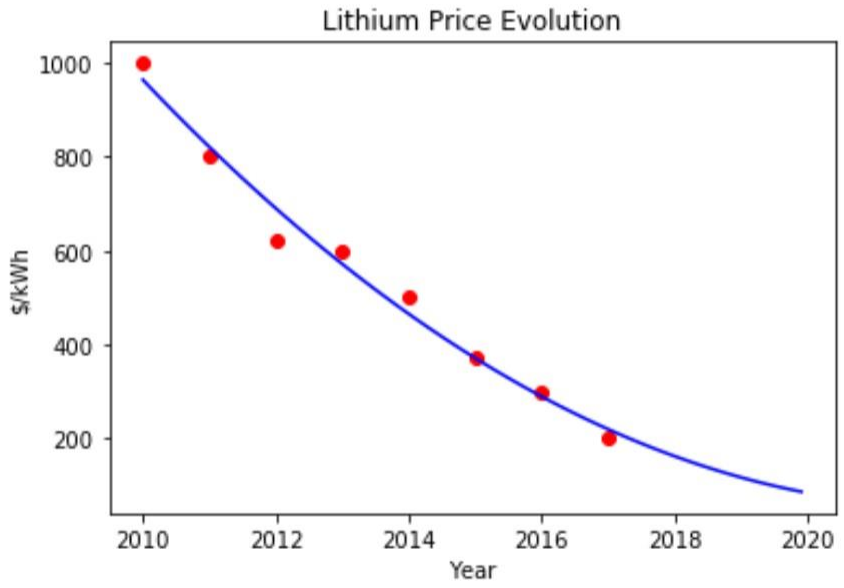
Demand for battery raw materials



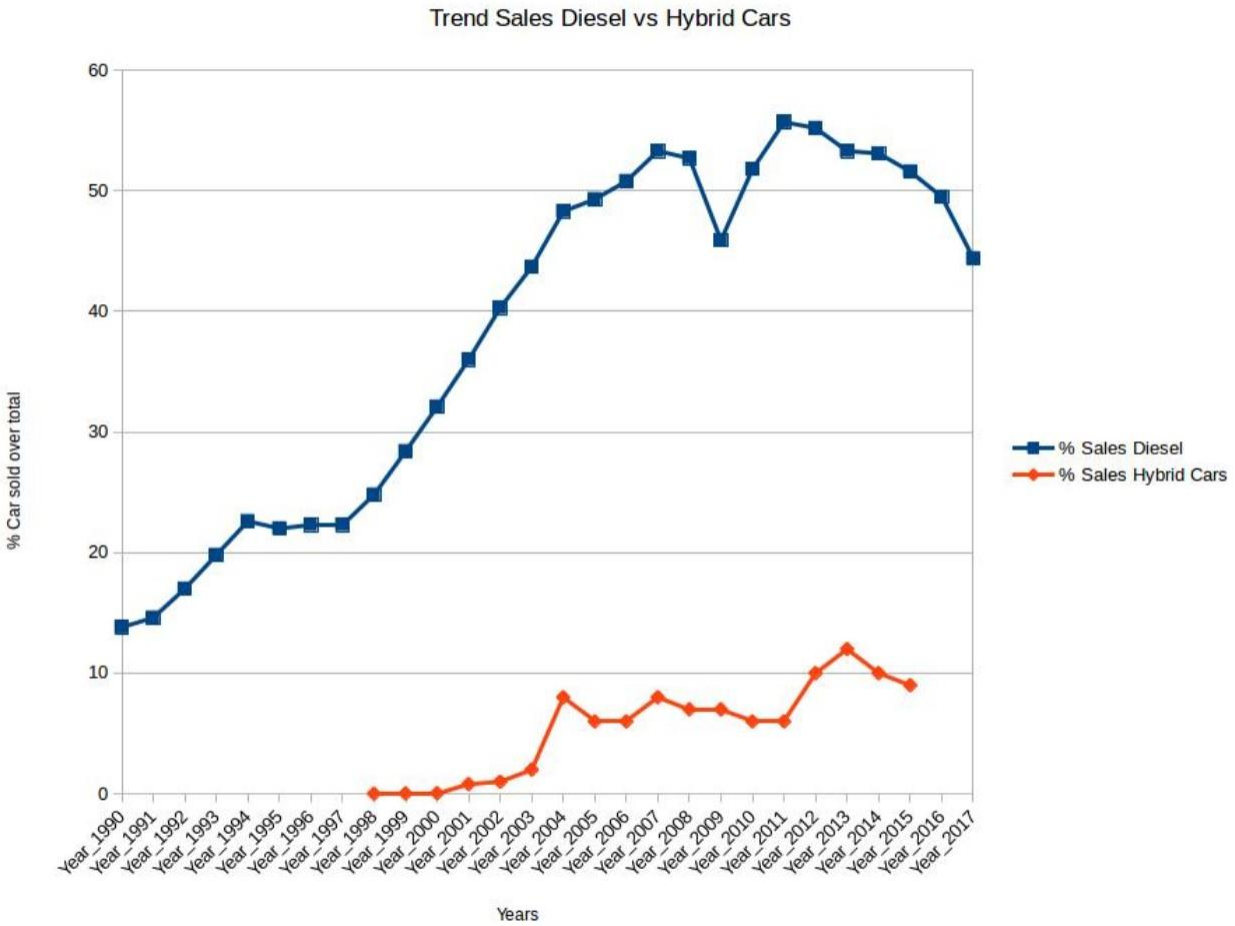
Source: Electric Vehicle Outlook 2018, Bloomberg New Energy Finance. Note: Copper includes copper current collectors and pack wiring. Aluminium includes aluminium current collectors, cell and pack materials and aluminium in cathode active materials.

Cobalt supply as one of the risk factors of EV adoption

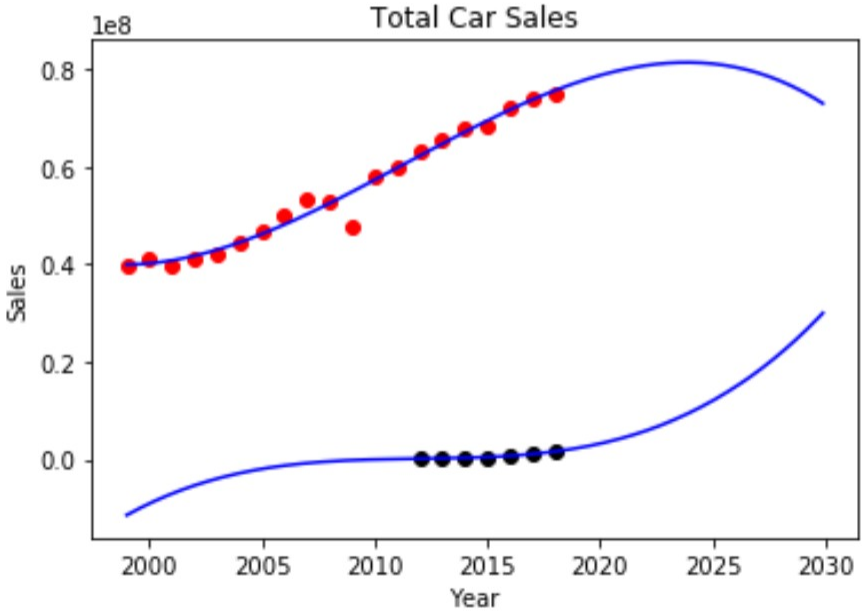
Battery price evolution



Switch to MNC batteries containing less cobalt to reduce risk and price

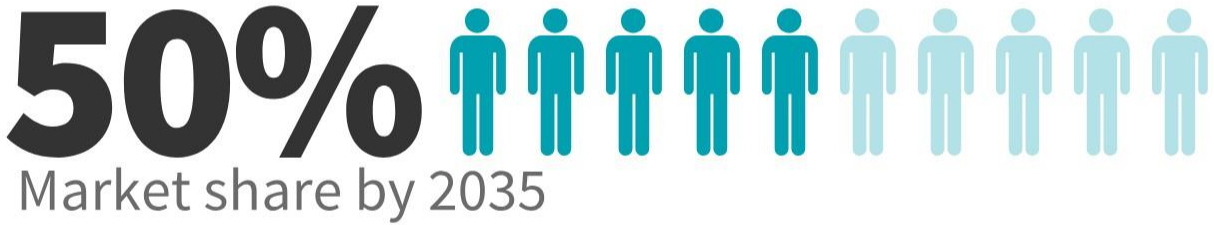

Trend Sales Diesel vs Hybrid Cars
% Car sold over total
60
50
40
30
20
10
0
Year_1990
Year_1991
Year_1992
Year_1993
Year_1994
Year_1995
Year_1996
Year_1997
Year_1998
Year_1999
Year_2000
Year_2001
Year_2002
Year_2003
Year_2004
Year_2005
Year_2006
Year_2007
Year_2008
Year_2009
Year_2010
Year_2011
Year_2012
Year_2013
Year_2014
Year_2015
Year_2016
Year_2017
Years
% Sales Diesel
% Sales Hybrid Cars

Global Electric vehicle sales prediction



- “Red” ICE car global sales
- “Black” EV sales
- EV evolution similar to other technologies (diesel + hybrid)

Conclusions



In [105]:
```
# Fitting Polynomial Regression to the EV dataset
poly_reg_BEV = PolynomialFeatures(degree = 3)
X_poly_BEV = poly_reg_BEV.fit_transform(X_BEV)
poly_reg_BEV.fit(X_poly_BEV, y_BEV)
lin_reg_BEV = LinearRegression()
lin_reg_BEV.fit(X_poly_BEV, y_BEV)
```

Out[105]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)
```

In [15]:
```
#########################
###  Plotting the data ###
#########################
```

In [119]:
```
# Visualising the Polynomial Regression results (for higher resolution and smoother curve)
X_grid = np.arange(min(X_normal), 2030, 0.1)
X_grid = X_grid.reshape((len(X_grid), 1))
plt.scatter(X_normal, y_normal, color = 'red')
plt.scatter(X_BEV, y_BEV, color = 'black')
plt.plot(X_grid, lin_reg_normal.predict(poly_reg_normal.fit_transform(X_grid)), color = 'blue')
plt.plot(X_grid, lin_reg_BEV.predict(poly_reg_BEV.fit_transform(X_grid)), color = 'blue')
plt.title('Total Car Sales')
plt.xlabel('Year')
plt.ylabel('Sales')
plt.show()
```