LAB_B1

February 9, 2016

1 Stratification of the solar atmosphere

1.1 FALC temperature stratification

Figure 2 shows the FALC temperature stratification. It was made with IDL code similar to the following. Write similar code and make it work.

```
In [1]: %matplotlib inline
        import numpy as np
        import matplotlib.pyplot as plt
        import matplotlib as mpl
        import astropy.constants as const

        #use the pretty LaTeX fonts
        mpl.rcParams.update({'text.usetex': True})
        plt.rc('font', family='serif', size=10)
        mpl.rc('axes.formatter', useoffset=False)

        # plt.style.use('ggplot')
```

```
In [2]: k = const.k_B.cgs.value
```

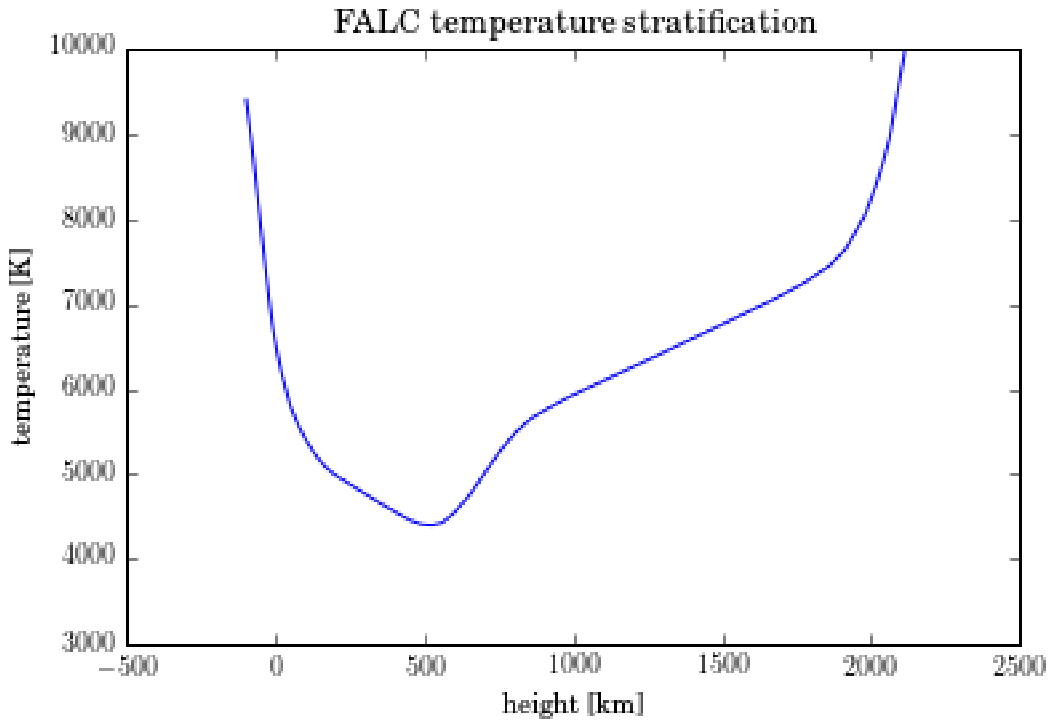
```
In [3]: !head falc.dat
```

```
FALC solar model atmosphere of Fontenla, Avrett & Loeser 1993ApJ...406..319F; 82 heights top-to-bottom
 height    tau_500    colmass     temp     v_turb n_Htotal    n_proton    n_electron pressure  p_gas/p  densit
 [km]      dimless    [g/cm^2]    [K]      [km/s]  [cm^-3]     [cm^-3]     [cm^-3]    [dyn/cm2] ratio     [g/cm

 2218.20  0.000E+00  6.777E-06  100000  11.73  5.575E+09  5.575E+09  6.665E+09  1.857E-01  0.952  1.306
 2216.50  7.696E-10  6.779E-06   95600  11.65  5.838E+09  5.837E+09  6.947E+09  1.857E-01  0.950  1.368
 2214.89  1.531E-09  6.781E-06   90816  11.56  6.151E+09  6.150E+09  7.284E+09  1.858E-01  0.948  1.441
 2212.77  2.597E-09  6.785E-06   83891  11.42  6.668E+09  6.667E+09  7.834E+09  1.859E-01  0.945  1.562
 2210.64  3.754E-09  6.788E-06   75934  11.25  7.381E+09  7.378E+09  8.576E+09  1.860E-01  0.941  1.729
 2209.57  4.384E-09  6.790E-06   71336  11.14  7.864E+09  7.858E+09  9.076E+09  1.860E-01  0.938  1.843
```

```
In [4]: # read in the data
        h, tau5, colm, temp, vturb, nhyd, nprot, nel, ptot, pgasptot, dens = np.loadtxt('falc.dat',skip

        hsun = h
```

```
In [5]: fig, ax = plt.subplots()
        ax.plot(h,temp)
        ax.set_ylim(3000,10000)
        ax.set_xlabel('height [km]')
```

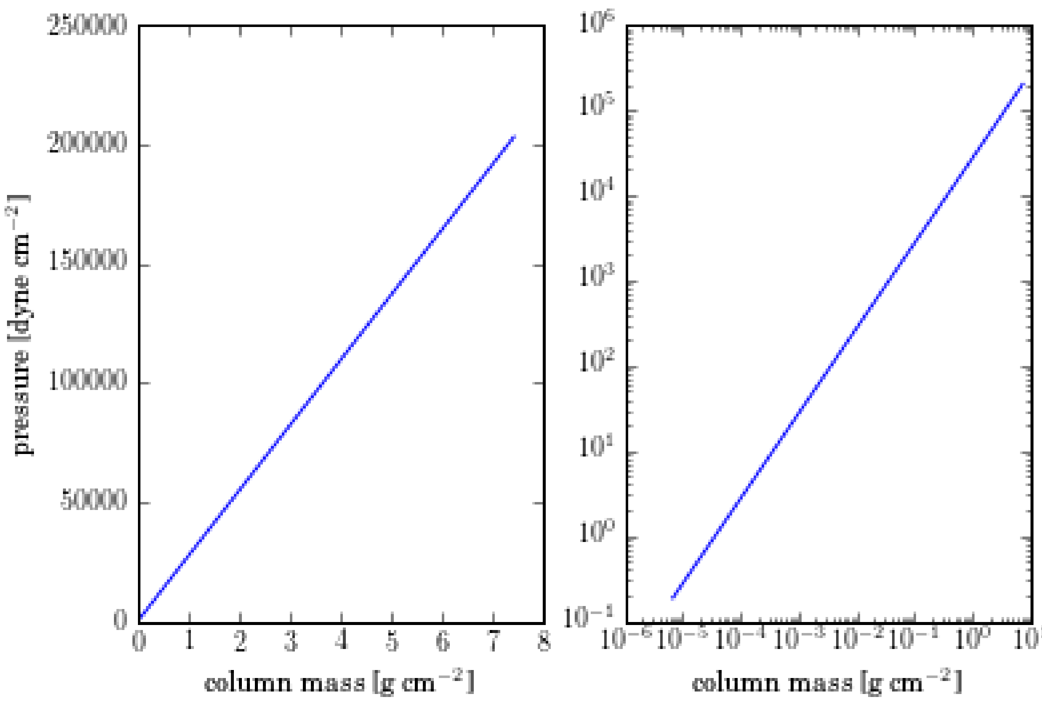
```
ax.set_ylabel('temperature [K]')
ax.set_title('FALC temperature stratification')
plt.show()
```



1.2 FALC density stratification

Plot the total pressure p_{total} against the column mass m, both linearly and logarithmically. You will find that they scale linearly. Exlpain what assumption has caused $p_{total} = cm$ and determine the value of solar surface gravity $g = c$ that went into the FALC-producing code.

```
In [6]: fig, ax = plt.subplots(ncols=2)

        ax[0].plot(colm,ptot)
        ax[0].set_xlabel('column mass [g cm$^{-2}$]')
        ax[0].set_ylabel('pressure [dyne cm$^{-2}$]')

        ax[1].loglog(colm,ptot)
        ax[1].set_xlabel('column mass [g cm$^{-2}$]')

        plt.show()
```



We assumed hydrostatic equilibrium.

$$\frac{P}{h} = \rho g = \frac{mg}{V} = \frac{m_{col}g}{h} \rightarrow P = gm$$

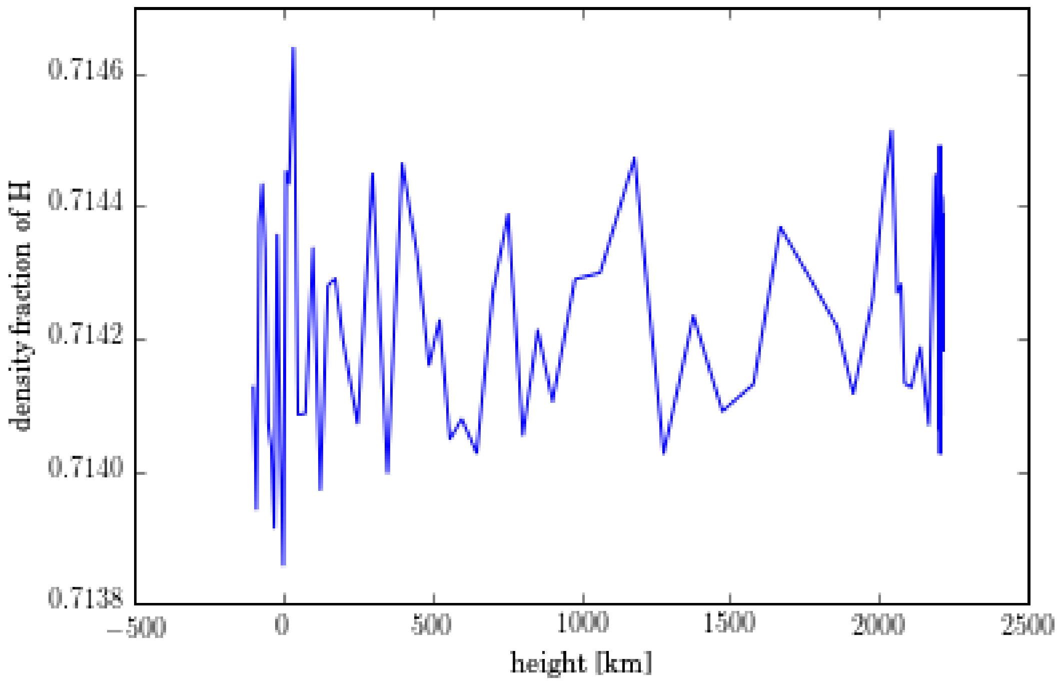
And $g = 27396.5 \rightarrow logg = 4.4$

```
In [7]: g = (ptot[-1] - ptot[0])/(colm[-1] - colm[0])
        print "solar g, logg = ",g, np.log10(g)
```

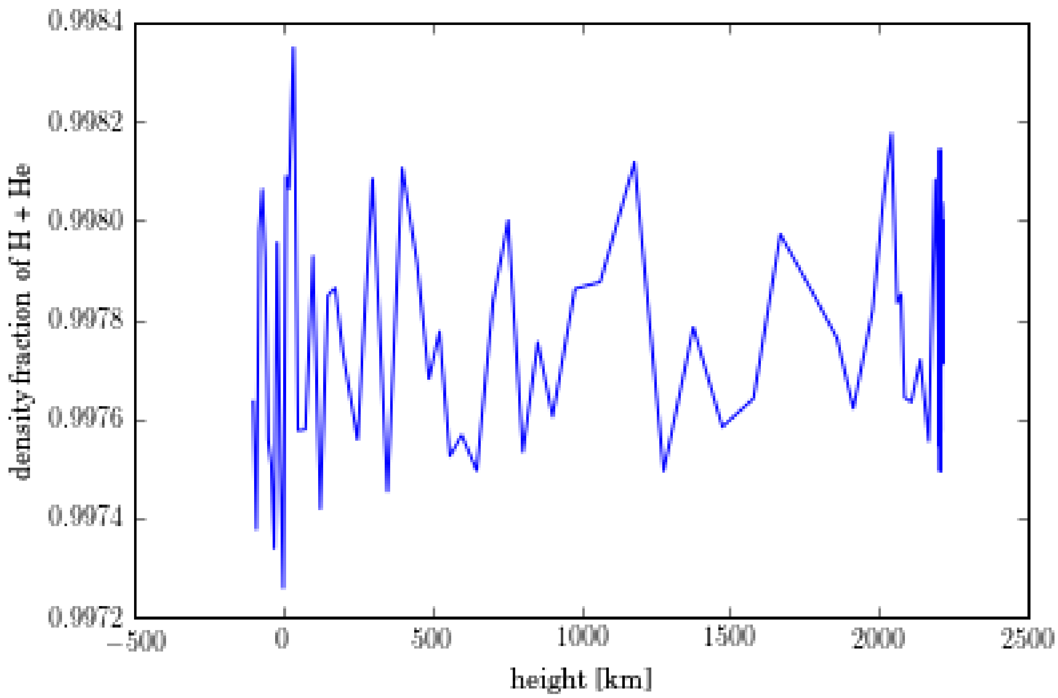
```
solar g, logg =  27396.5215001 4.43769542453
```

Fontenla et al (1993) also assumed complete mixing, i.e., the same element mix at all heights. Check this by plotting the ratio of the hydrogen mass density to the total mass density against height. Then add helium to hydrogen using their abundance and mass ratios ($N_{He}/N_H = 0.1, m_{He} = 3.97 m_H$), estimate the fraction of the total mass density made up by the remaining elements in the model mix ("the metals").

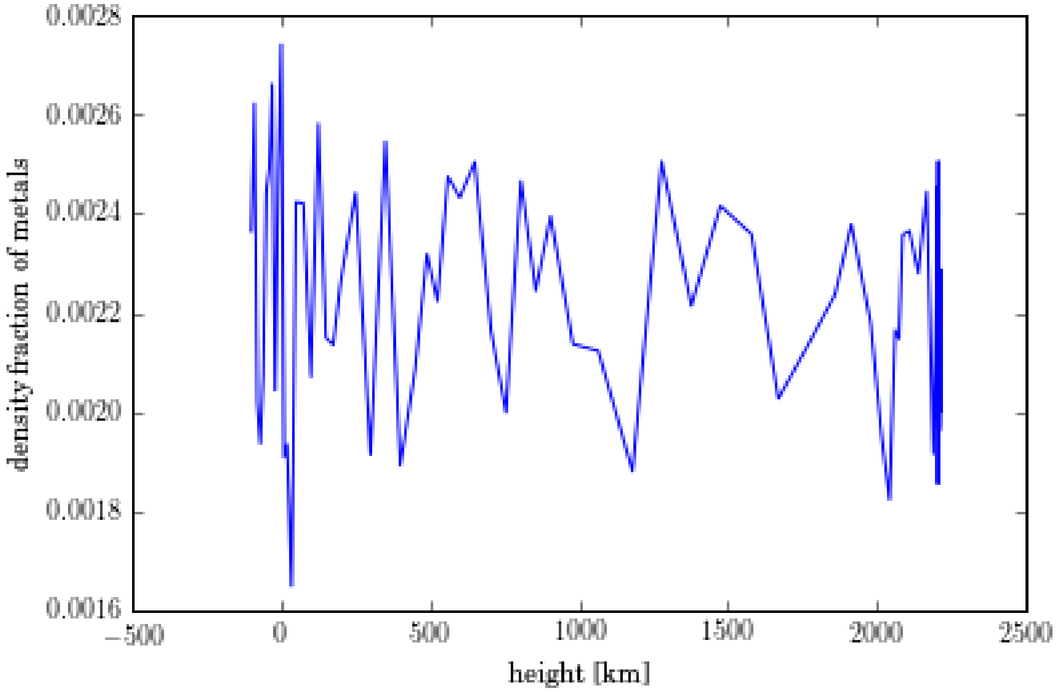
```
In [8]: mh = 1.67352e-24 # mass of hydroen [g]
        H_ratio = nhyd*mh / dens
        fig, ax = plt.subplots()
        ax.plot(h,H_ratio, label='H')
        ax.set_ylabel('density fraction of H')
        ax.set_xlabel('height [km]')

        plt.show()
```



In [9]:
```
fig, ax = plt.subplots()
HHe_ratio = (nhyd*mh + 0.1*nhyd*3.97*mh) / dens
ax.plot(h, HHe_ratio)
ax.set_ylabel('density fraction of H + He')
ax.set_xlabel('height [km]')
plt.show()
```



```
In [10]: fig, ax = plt.subplots()
         metal_ratio = 1 - (nhyd*mh + 0.1*nhyd*3.97*mh)/dens
         ax.plot(h, metal_ratio)
         ax.set_ylabel('density fraction of metals')
         ax.set_xlabel('height [km]')
         plt.show()
```

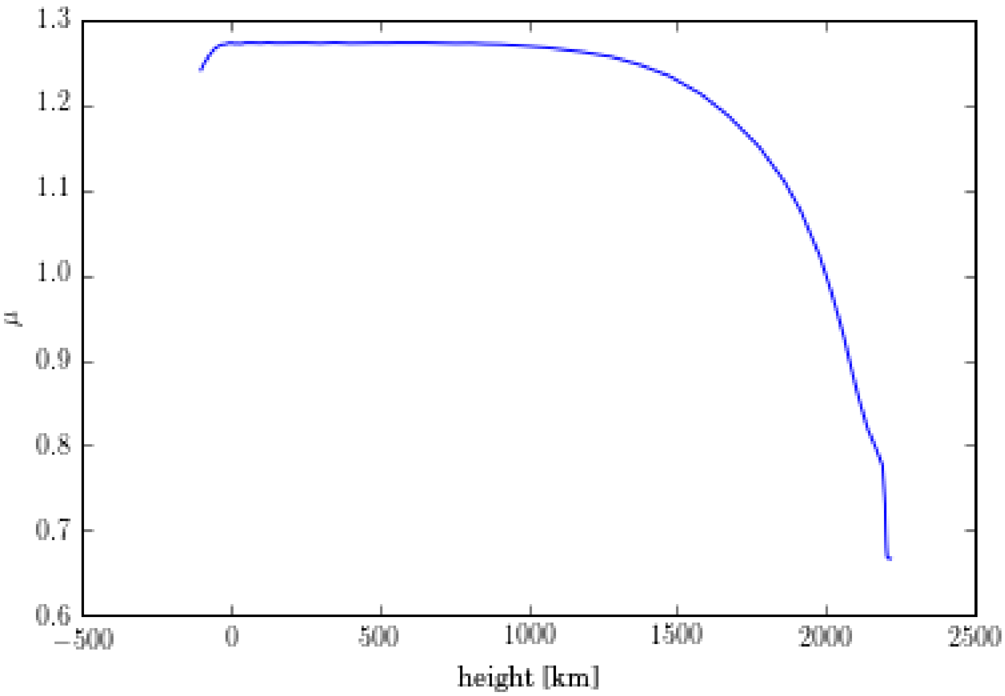



```
In [11]: print(r"average metal fraction is ~ {:.2}".format((1 - HHe_ratio).mean()))

average metal fraction is ~ 0.0022
```

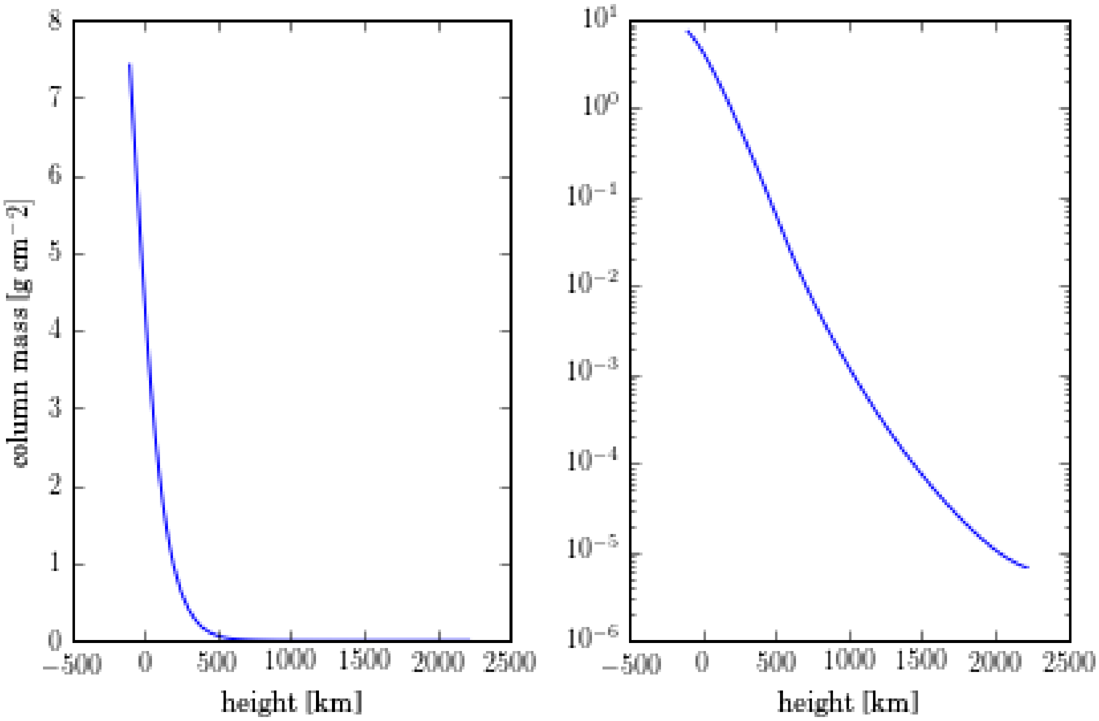
Plot the column mass against height. The curve becomes nearly straight when you make the y-axis logarithmic. Why is that? Why isn't it exactly straight?

It is neary straight since via hydrostaic equilibrium $log(m_{col}) = -h\mu g/kT$ but it depends on μ which changes slightly at different depths in the atmosphere as seen below.

```
In [12]: mu = dens/((nhyd+0.1*nhyd+nprot)*mh)
         plt.plot(h,mu)
         plt.xlabel('height [km]')
         plt.ylabel(r'$\mu$')

Out[12]: <matplotlib.text.Text at 0x106a74450>
```

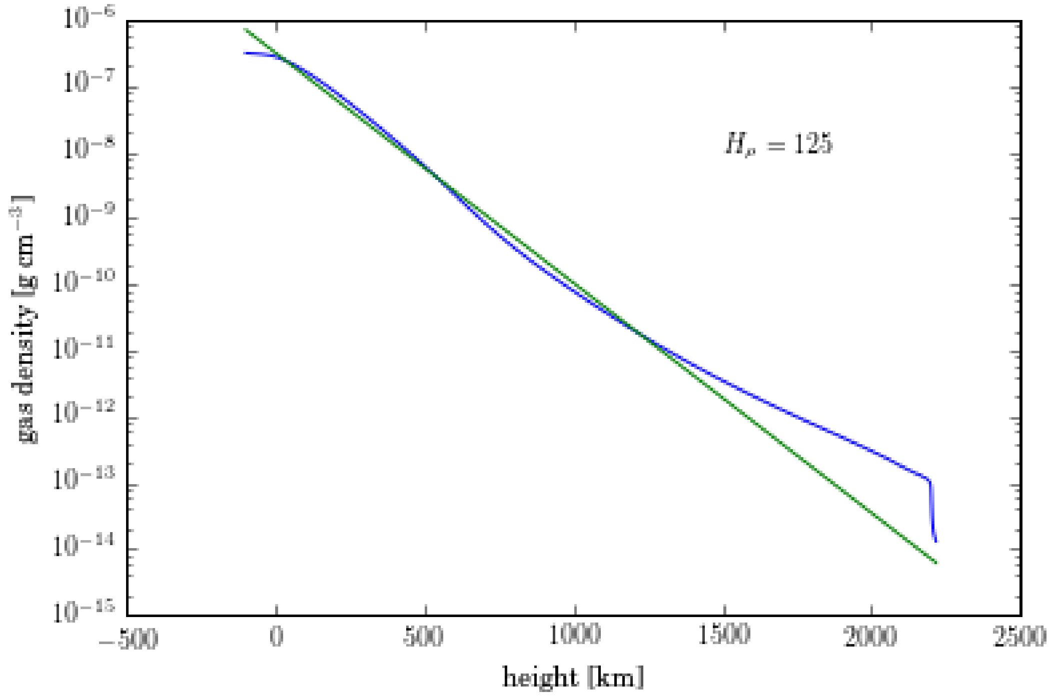



```
In [13]: fig, ax = plt.subplots(ncols=2)
         ax[0].plot(h, colm)
         ax[0].set_xlabel('height [km]')
         ax[0].set_ylabel('column mass [g cm^-2]')
         ax[1].semilogy(h, colm)
         ax[1].set_xlabel('height [km]')
         fig.tight_layout()
         plt.show()
```



Plot the gas density against height. Estimate the density scale height H_ρ in $\rho \approx \rho(0)exp(-h/H_\rho)$ in the photosphere.

```
In [14]: def rho(h, rho0, H):
             return rho0*np.exp(-h/H)

         plt.semilogy(h, dens)
         plt.semilogy(h, rho(h, dens[79], 125))
         plt.text(1500,1e-8, r"$H_{\rho} = 125$")
         plt.ylabel('gas density [g cm$^{-3}$]')
         plt.xlabel('height [km]')
         plt.show()
```

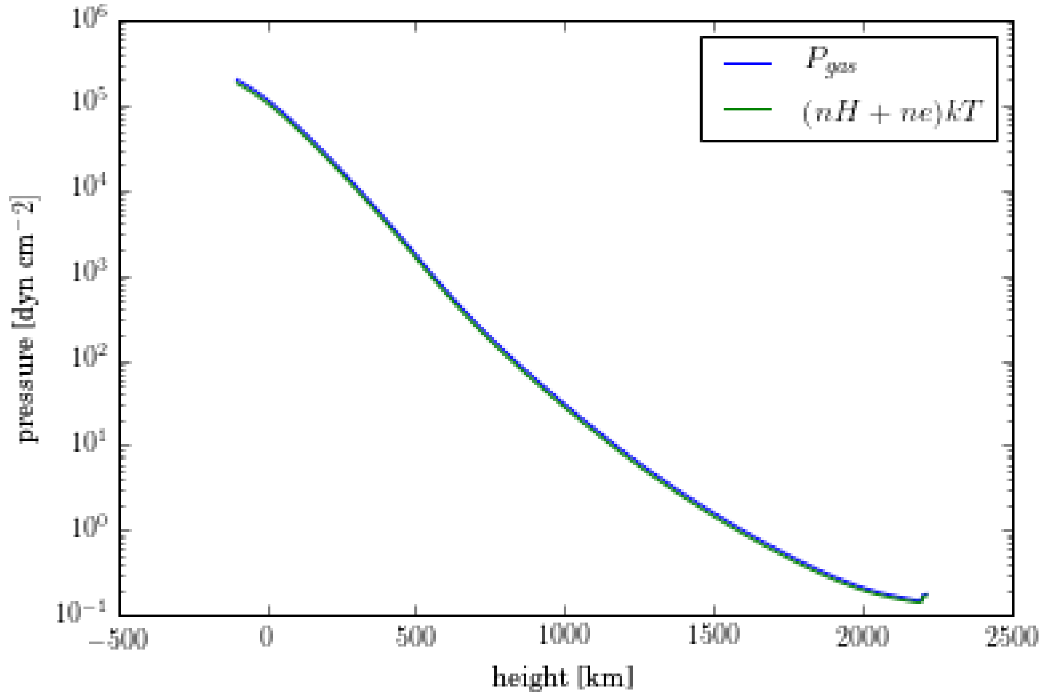


scale height $H_\rho \approx 125$

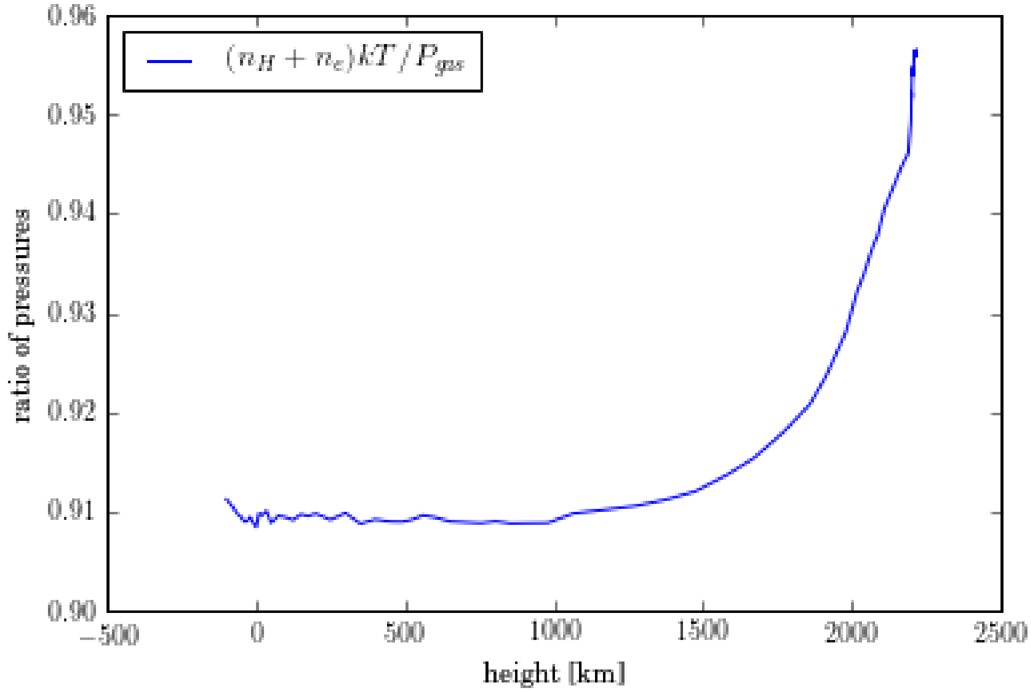
Compute the gas pressure and plot it against height. Overplot the product $(nH+ne)kT$. Plot the ratio of the two curves to show their differences. Do the differences measure deviations from the ideal gas law or something else? Now add the helium density NHe to the product and enlarge the deviations. Comments?

The differences are not deviations from the ideal gas law, rather they reflect the fact we arent including the contribution from He.

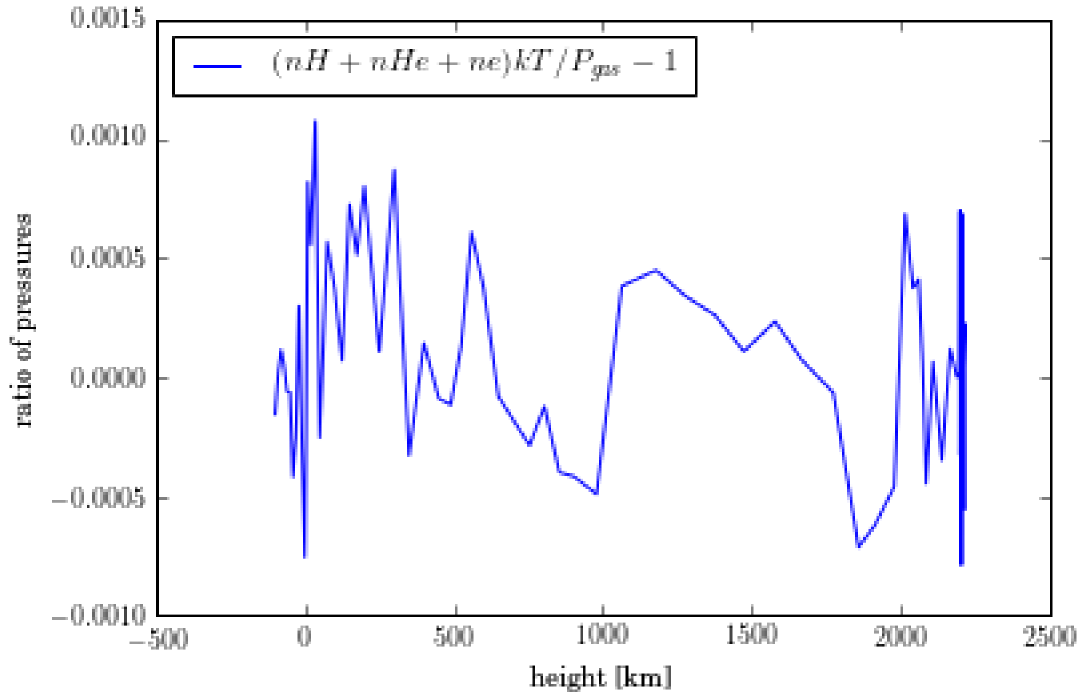
```
In [15]: #£ P_{gas} = P_{tot} - \rho v_{t}^2/2 £
         #pgas = ptot - dens*vturb**2/2
         pgas = pgasptot*ptot
         plt.semilogy(h, pgas, label=r'$P_{gas}$')
         plt.semilogy(h, (nhyd+nel)*k*temp, label = r'$(nH+ne)kT$' )
         plt.xlabel('height [km]')
         plt.ylabel('pressure [dyn cm^-2]')

         plt.legend()
         plt.show()
```




```
In [16]: p2 = (nhyd+nel)*k*temp
         p3 = (nhyd+nel+0.1*nhyd)*k*temp

         plt.plot(h, p2/pgas, label=r'$(n_H+n_e)kT/P_{gas}$')
         plt.xlabel('height [km]')
         plt.ylabel('ratio of pressures')

         plt.legend(loc='upper left')
         plt.show()
```




```
In [17]: p3 = (nhyd + nel + 0.1*nhyd)*k*temp
         plt.plot(h, p3/pgas-1, label=r'$(nH+nHe+ne)kT/P_{gas}-1 $')

         plt.xlabel('height [km]')
         plt.ylabel('ratio of pressures')

         plt.legend(loc='upper left')
         plt.show()
```

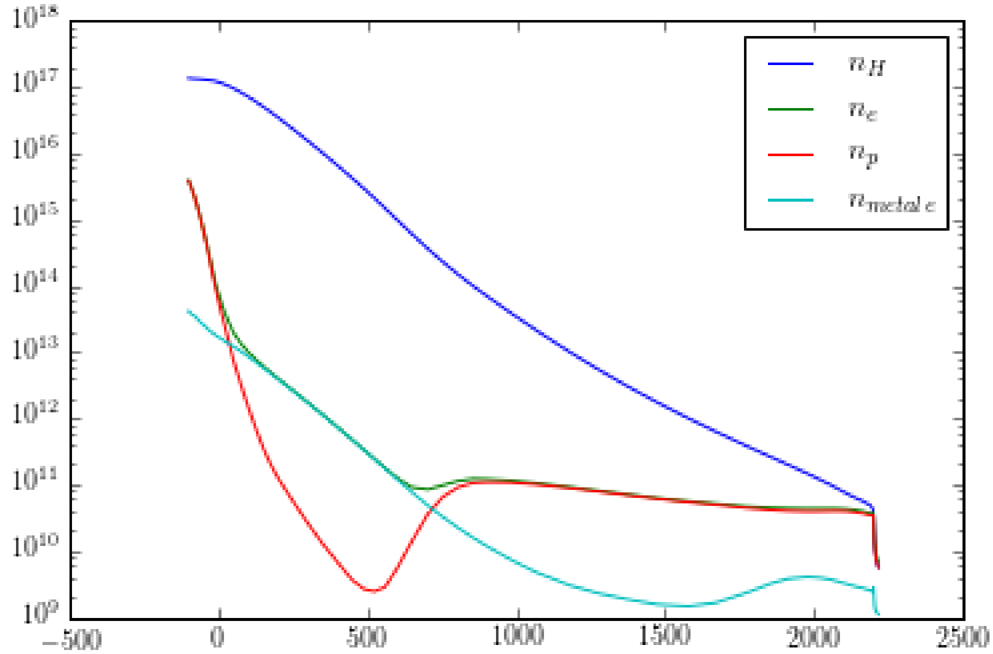


Comments?

The addition of the He density puts the gas pressure in agreement with the ideal gas law.

Plot the total hydrogen density against height and overplot curves for the electron density, the proton density, and the density of the electrons that do not result from hydrogen ionization. Explain their behavior. You may find inspiration in Figure 6 on page 13. The last curve is parallel to the hydrogen density over a considerable height range. What does that imply? And what happens at larger height?

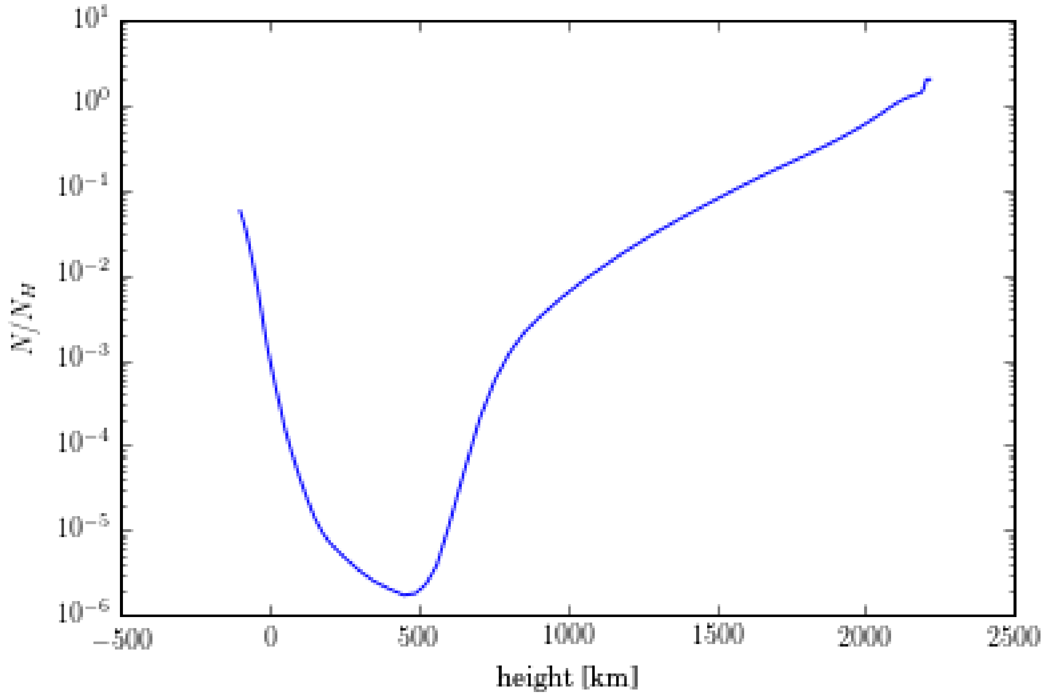
In [18]:
```
plt.semilogy(h, nhyd, label=r'$n_H$')
plt.semilogy(h, nel, label=r'$n_e$')
plt.semilogy(h, nprot, label=r'$n_p$')
n_metal = nel - nprot
plt.semilogy(h, n_metal, label=r'$n_{metal e}$')
plt.legend()
plt.show()
```



The n_p traces the temperature as it decreases then inverts. From $h = 0$ until the inversion, n_e is dominated by electrons coming from ionized metals. At the inversion, more H becomes ionized and this becomes the dominant electron source. The density of electrons due to metal ionization paralles the density of hydrogren since n_H traces the density of metals through out the atmosphere, which exponentially decreases with height. At larger heights up in the atmosphere the temperature rises enough to further ionization of the metals, increaseing the number of electrons (compared to the number of metal ions) and flattening out the n_{metale^-}curve.

Plot the ionization fraction of hydrogen logarithmically against height. Why does this curve look like the one in Figure 2? And why is it tilted with respect to that?

In [19]:
```
N = nprot + nel - n_metal
plt.semilogy(h, N/nhyd, label=r'$n_H$')
plt.xlabel('height [km]')
plt.ylabel('$N/N_H$')
plt.show()
```



The ionization fraction mirrors the temperature stratification since the temperature sets the ionization rate given by the Saha equation which scales as $T^{3/2}e^{-T}$. This follows T until very high temperatures in the outer atmosphere where the exponential takes over and the ionization fraction diverges from the temperature curve.

** Let us now compare the photon and particle densities. In thermodynamic equilibrium (TE) the radiation is isotropic with intensity $I_\nu = B_\nu$ and has total energy density (Stefan Boltzmann)**

$$u = \frac{1}{c}\iint B_\nu d\Omega d\nu = \frac{4\sigma}{c}T^4$$

so that the total photon density for isotropic TE radiation is given, with $u_\nu = du/d\nu$, T in K and N_{phot} in photons per cm^3, by

$$N_{phot} = \int_0^\infty \frac{u_\nu}{h\nu}d\nu \approx 20T^3$$

This equation gives a reasonable estimate for the photon density at the deepest model location, why?

We are assuming the deepest levels are in LTE since the density is high enough for collisions to couple photons and Hydrogen.

Compute the value there and compare it to the hydrogen density.

$N_{phot} = 1.6 \times 10^{13}$ and $N_{phot}/N_H = 0.00012$

```
In [20]: n = 20*temp[-1]**3
         #print n, nhyd[-1], temp[-1], temp[0]
         print "{:.2}".format(n/nhyd[-1])

         natmo = 20*5770**3/(2*np.pi)

         print natmo, natmo/nhyd[0]
```

```
0.00012
6.11473396401e+11 109.681326709
```

Why is the equation not valid higher up in the atmosphere? It is not valid higher up in the atmosphere because the density is low enough that we are in the NLTE regime.

The photon density there is $N_{phot} \approx 20T_{eff}^3/2\pi$ with $T_{eff} = 5770$K the effective solar temperature (since $\pi B(T_{eff}) = \sigma T_{eff}^3 = F^+ = \pi\overline{I^+}$ with F^+ the emergeng flux and $\overline{I^+}$ the disk averaged emergent intensity). Compare it to the hydrogen density at the highest location in the FALC model. The medium there is insensitive to these photons (except those at the center wavelength of the hydrogen Ly α line), why?

$N_{phot} = 6 \times 10^{11}$ and $N_{phot}/N_H = 110$

The medium is insensitive to these photons since the optical depth outside of line center is $<< 1$ (the pressure is low so there is no pressure broadening) and thus we are in NLTE due to the low density at these altitudes.

1.3 Comparison with the earth's atmosphere

Write IDL code to read file earth.dat.

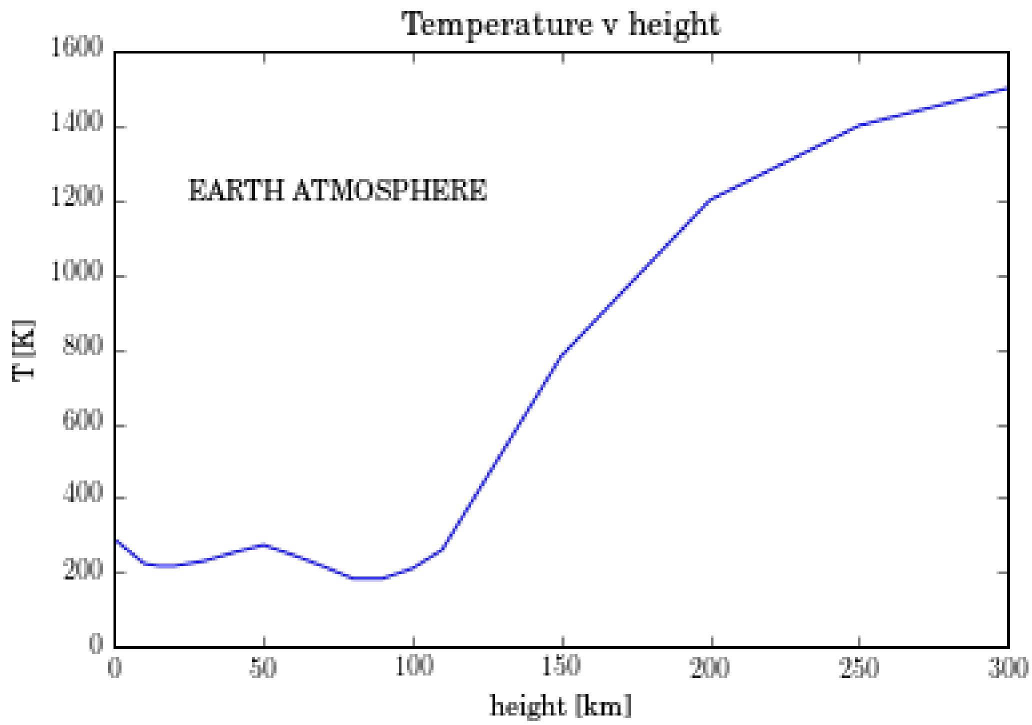
```
In [21]: !head earth.dat
```

```
0   6.01   288   -2.91   19.41
   1   5.95   282   -2.95   19.36
   2   5.90   275   -3.00   19.31
   3   5.85   269   -3.04   19.28
   4   5.79   262   -3.09   19.23
   5   5.73   256   -3.13   19.19
   6   5.67   249   -3.18   19.14

   8   5.55   236   -3.28   19.04
```

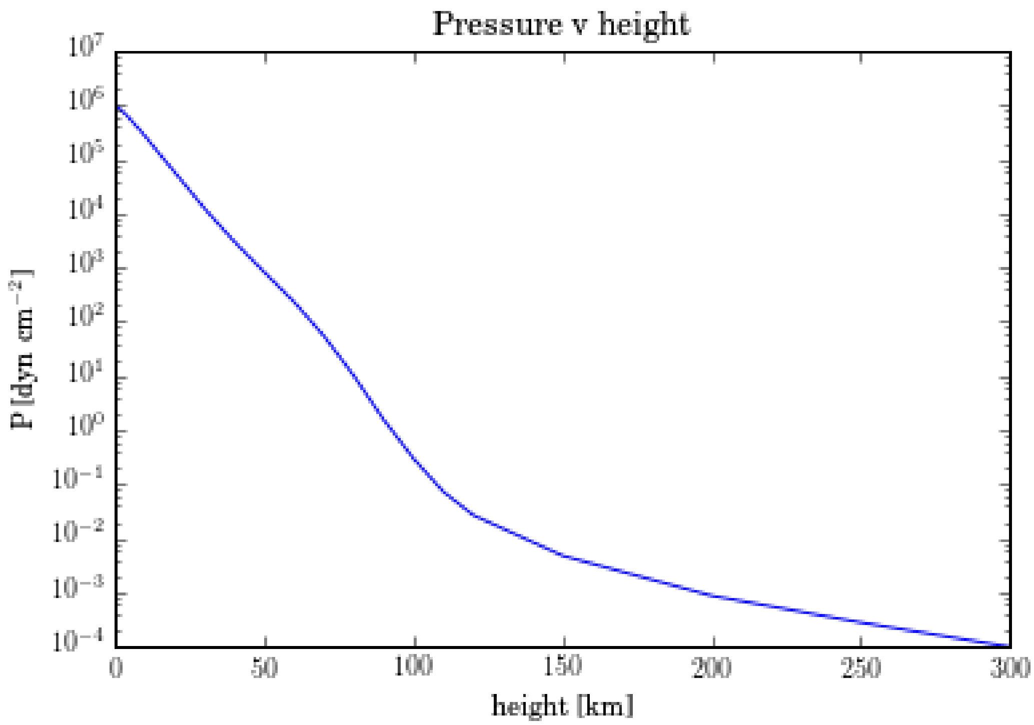
```
In [22]: h, logP, T, logdens, logN = np.loadtxt('earth.dat', unpack=True)
```

Plot the temperature, pressure, particle density and gas density against height, logarithmically where appropriate.

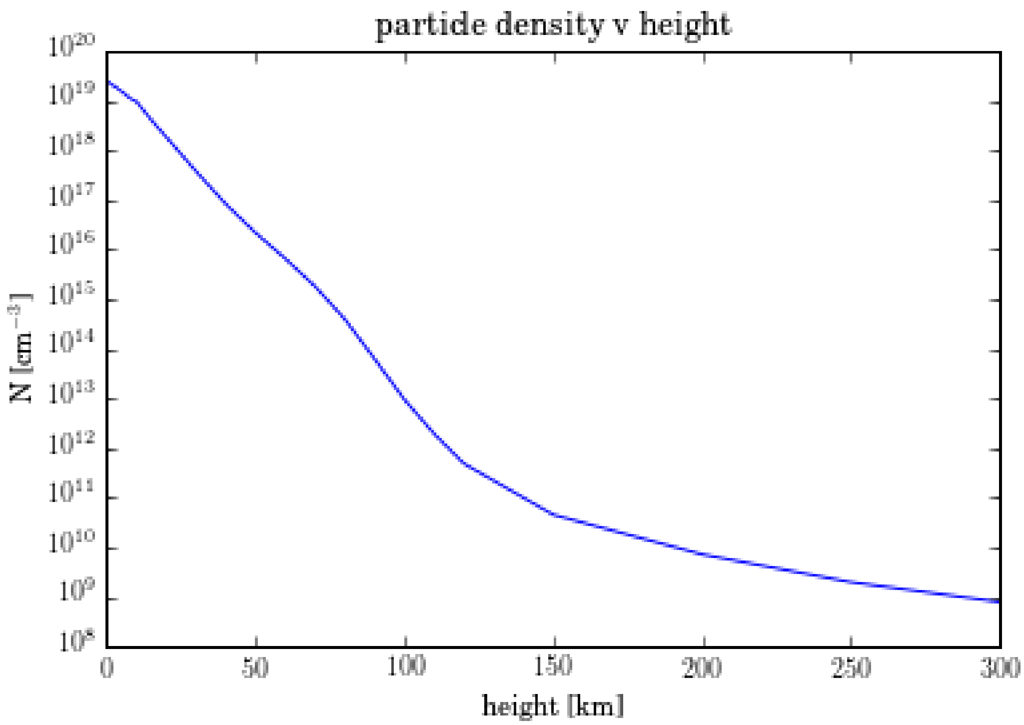
```
In [23]: plt.plot(h, T)
         plt.ylabel('T [K]')
         plt.xlabel('height [km]')
         plt.title('Temperature v height')
         plt.text(25, 1200, 'EARTH ATMOSPHERE')
         plt.show()
```



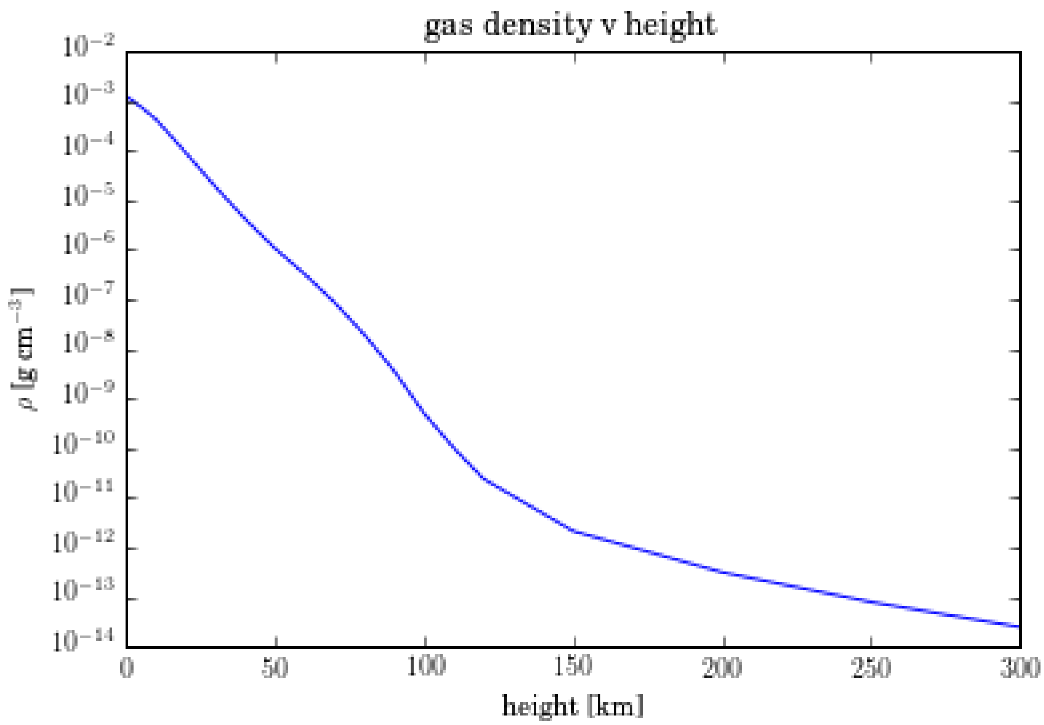

```
In [24]: P = 10**logP
         plt.semilogy(h, P)
         plt.ylabel('P [dyn cm$^{-2}$]')
         plt.title('Pressure v height')
         plt.xlabel('height [km]')
         plt.show()
```




```
In [25]: N = 10**logN
         plt.semilogy(h, N)
         plt.ylabel('N [cm$^{-3}$]')
         plt.title('particle density v height')
         plt.xlabel('height [km]')
         plt.show()
```

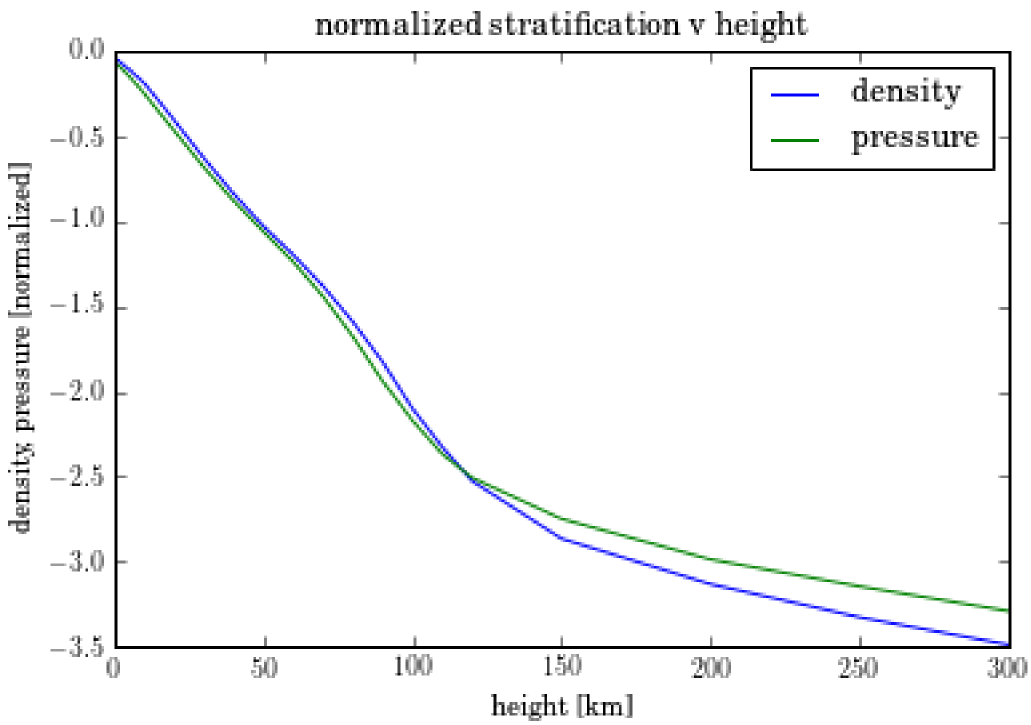



```
In [26]: dens = 10**logdens
         plt.semilogy(h, dens)
         plt.ylabel(r'$\rho$ [g cm$^{-3}$]')
         plt.title('gas density v height')
         plt.xlabel('height [km]')
         plt.show()
```



Plot the pressure and density stratifications together in normalized units in one graph. Comments?

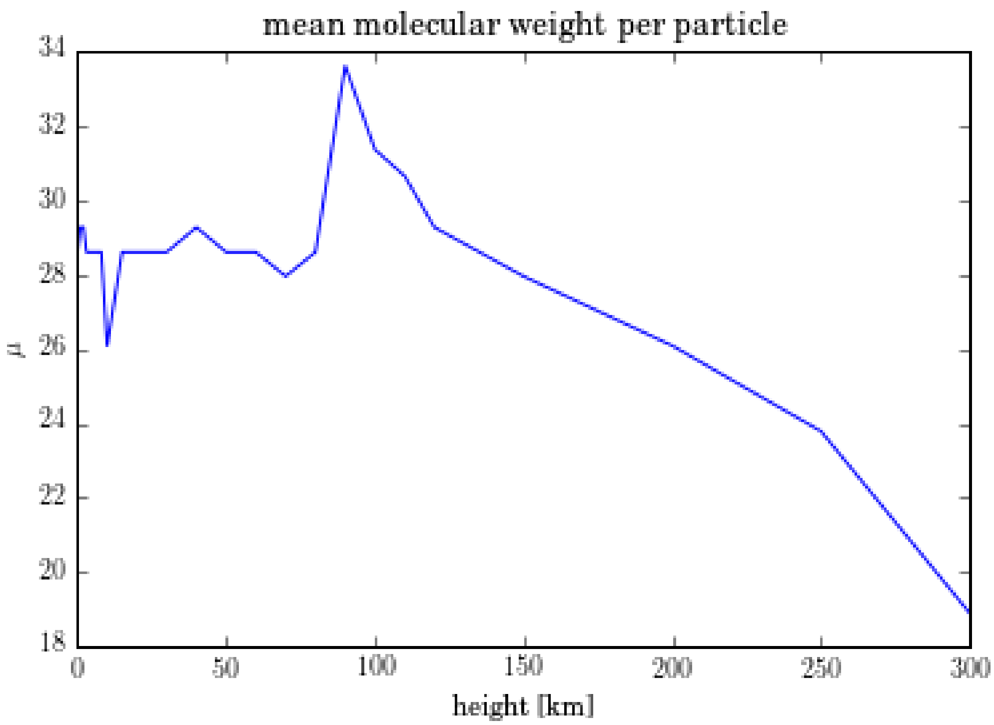
In [27]:
```
# normalze:

norm = (logP[0] + logdens[0])


plt.plot(h, logdens/norm+0.9, label='density')
plt.plot(h, logP/norm-2.0, label = 'pressure')
plt.ylabel(r'density, pressure [normalized]')
plt.title('normalized stratification v height')
plt.xlabel('height [km]')
plt.legend()
plt.show()
```



The pressure and density follow each other for most of the height of the atmosphere. However, they depart at around heights around 120km. This departure must be due to a difference in species at higher heights acording to the ideal gas law, $P = (N/V)kT = (\rho/\mu)kT$.

Plot the mean molecular weight $\mu_E = \overline{m}/m_H = \rho/(N m_H)$ against height. Why does it decrease in the high atmosphere?

```
In [28]: # h, logP, T, logdens, logN

         dens = 10**(logdens)
         N = 10**(logN)

         mu = dens/(N*mh)

In [29]: plt.plot(h, mu)
         plt.ylabel(r'$\mu$')
         plt.title('mean molecular weight per particle')
         plt.xlabel('height [km]')
         plt.show()
```

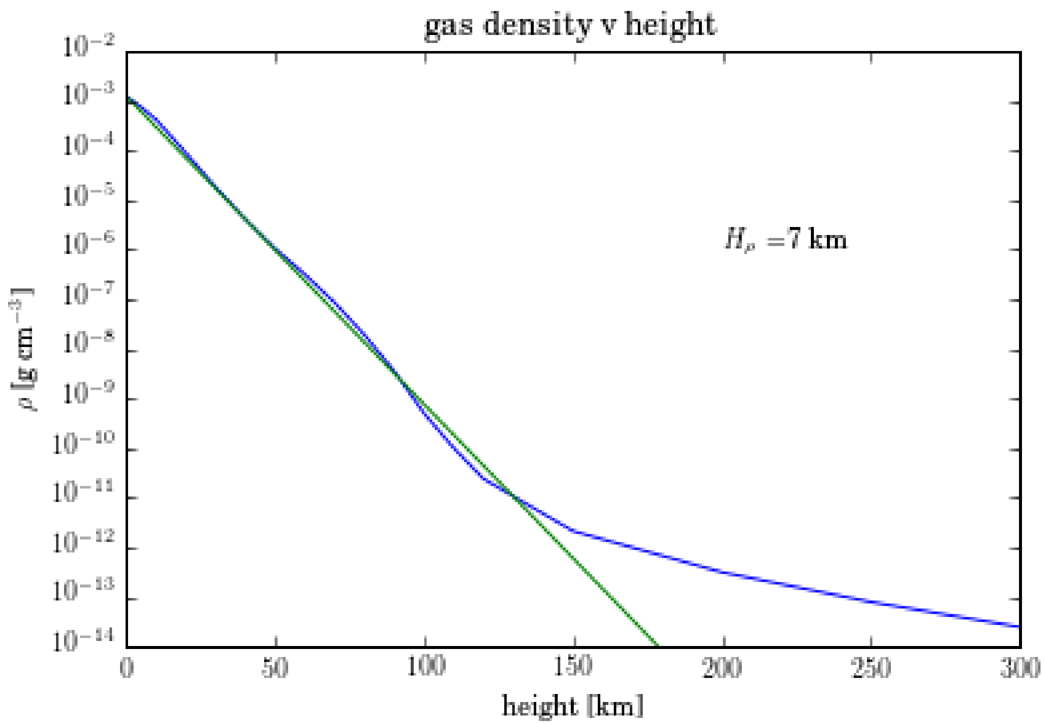


The mean molecular weight decreases high in the atmosphere because heavier molecules segregate to the lower atmosphere as only the lighter molecules have enough kinetic energy to reach those heights. In addition, as the height increases molecules may become dissasociated due to UV dissociation.

Estimate the density scales height of the lower terrestrial atmosphere. Which quantities make it differ from the solar one? How much harder do you have to breathe on Mount Everest?

The density profile makes the earth scale height differ from the solar scale height.

```
In [30]: plt.semilogy(h, dens)
         plt.semilogy(h, rho(h, dens[0], 7))
         plt.ylabel(r'$\rho$ [g cm$^{-3}$]')
         plt.title('gas density v height')
         plt.xlabel('height [km]')
         plt.ylim(1e-14, 1e-2)
         plt.text(200,1e-6, r"$H_{\rho} = $7 km")
         plt.show()
```




```
In [31]: print "density at summit",dens[8] / dens[0]
         print "pressure at summit",10**logP[8] / 10**logP[0]
```

```
density at summit 0.338844156139
pressure at summit 0.257039578277
```

Everest is ˜9000m ˜10km so the gas density is ˜1/3 (slightly less than $1/e$) the density at sea level while the pressure is 1/4 that of the surface. Therefore one would have to breath ˜3 times harder to get the same amount of oxygen.

Compare the terrestrial parameter values to the solar ones, at the base of each atmosphere. What is the ratio of the particle densities at h = 0 in the two atmospheres?

```
In [32]: # terrestrial: h, logP, T, logdens, logN
         # solar: hsun, ptot, temp, dens, nhyd
```

```
In [33]: print "      solar \t earth"
         print "h     {:.2} \t {:.2}".format(hsun[-11], h[0])
         print "press {:.2} \t {:.2}".format(ptot[-11], 10**logP[0])
         print "temp  {:.2} \t {:.2}".format(temp[-11], T[0])
         print "dens  {:.2} \t {:.2}".format(dens[-11], 10**logdens[0])
         print "N     {:.2} \t {:.2}".format(nhyd[-11], 10**logN[0])
         print '\n'
         print "ratio of particle density N_earth/N_sun {:.4}".format(10**logN[0]/nhyd[-11])
```

```
solar           earth
h     0.0         0.0
press 1.2e+05           1e+06
temp  6.5e+03           2.9e+02
```

```
dens  3.2e-07          0.0012
N     1.2e+17          2.6e+19


ratio of particle density N_earth/N_sun 217.5
```

```
In [34]: mcol_earth = 10**logP[0]/980.665
         print mcol_earth
```

```
1043.46845486
```

The standard gravity at the earth's surface is $g_E = 980.665$ cm s^{-2}. Use this value to estimate the atmospheric column mass (g cm^{-2}) at the earth's surface and compare that also to the value at the base of the solar atmosphere.

using $P_{tot} = gm_{col} \rightarrow m_{col} = 1043.46g/cm^2$

Final question: the energy flux of the sunshine reaching our planet ("irradiance") is

$$R = \frac{4\pi R^2}{4\pi D^2} = F_\odot^+$$

with $F_\odot^+ = \pi B(T_{eff}^\odot)$ the emergent solar flux, R the solar radius and D the sun-earth distance so that the sunshine photon density at earth is

$$N_{phot} = \pi \frac{R^2}{D^2} N_{phot}^{top}$$

with N_{phot}^{top} the photon density at the top of FALC which you determined at the end of Section 1.1. Compare N_{phot} to the particle density in the air around us, and to the local thermal photon production derived from (2). Comments?

```
In [35]: R = const.R_sun.cgs.value
         D = const.au.cgs.value
         Teff = 5770
         Nphot_sun = (20*Teff**3)/(2*np.pi)
         Nphot_earth = np.pi * (R**2/D**2) * Nphot_sun

         Tearth = T[0]
         Nphot_from_earth = (20*Tearth**3)/(2*np.pi)
```

```
In [36]: print "photon density at earth from the sun: {:.2}".format(Nphot_earth)
         print "particle density around us: {:.2}".format((10**(logN[0])))
         print "local thermal photon production: {:.2}".format(Nphot_from_earth)
```

```
photon density at earth from the sun: 4.2e+07
particle density around us: 2.6e+19
local thermal photon production: 7.6e+07
```

The photon density is ~11 orders of magnitude less than the particle density at the surface of earth. In addition the local thermal photon production is greater than that recieved by the sun.