Exercise 11.1 (Normal transformation)

a)

$$
\begin{aligned}
n' \cdot t' = (n')^T t &\overset{!}{=} n^T t = n \cdot t \\
n^T G^T M t &\overset{!}{=} n^T t \\
\Rightarrow G^T M &= \mathbb{1} \\
G^T &= M^{-1} \\
\Rightarrow \qquad G &= \left(M^{-1}\right)^T
\end{aligned}
$$

b) The normal has to point outside, but if some components are inverted we could not assume that this is conserved.

c) View the ellipsoid as a dragged unit ball:

- ball
  $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ with $x^2 + y^2 + z^2 = 1$
- ellipsoid
  $\begin{pmatrix} ax \\ by \\ cz \end{pmatrix}$ with $\left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 + \left(\frac{z}{c}\right)^2 = 1$

In our case are $a = 4$, $b = 2$, $c = 1$

$\Rightarrow M = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

$\Rightarrow G = \left(M^{-1}\right)^T = \begin{pmatrix} \frac{1}{4} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$

normals of ball: $n_B = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$

normal od ellipsoid: $n_e = G n_B = \begin{pmatrix} \frac{x}{4} \\ \frac{y}{2} \\ z \end{pmatrix}$

Exercise 11.2 (Cox-De Boor algorithm)
Out of the Lecture, the definition of $B_{i,j}(t)$

$$B_{i,j}(t) = \tfrac{t-t_i}{t_{i+j}-t_i} B_{i,j-1}(t) + \tfrac{t_{i+j}-t}{t_{i+j}-t_i} B_{i,j-1}(t) \tag{1}$$

and $B_{i,0}(t) = \delta_{i,l}$ we could follow that $B_{i,j}(t) = 0$ if $i > l$ or $i + j < l$. Now lets show that

$$\sum_{i=l-d+k}^{l} B_{i,d-k}(t) p_i^k \overset{!}{=} \sum_{i=l-d+k+1}^{l} B_{i,d-k-1}(t) p_i^{k+1} \tag{2}$$

We start using 1

$$\sum_{i=l-d+k}^{l} B_{i,d-k}(t) p_i^k = \sum_{i=l-d+k}^{l} \left(\frac{t-t_i}{t_{i+d-k}-t_i} B_{i,d-k-1}(t) + \frac{t_{i+d-k+1}-t}{t_{i+d-k+1}-t_{i+1}} B_{i+1,d-k-1}(t) \right) p_i^k \tag{3}$$

$$= \sum_{i=l-d+k+1}^{l} B_{i,d-k-1}(t) \underbrace{\left(\frac{t-t_i}{t_{i+d-k}-t_i} p_i^k + \frac{t_{i+d-k}-t}{t_{i+d-k}-t_i} p_{i-1}^k \right)}_{p_i^{k+1}} \tag{4}$$

$$+ \frac{t-t_{l-d+k}}{t_l-t_{l-d+k}} \underbrace{B_{l-d+k,d-k-1}}_{=0\ \forall\, d,k}(t) p_{l-d+k}^k + \frac{t_{l+d-k+1}-t}{t_{l+d-k+1}-t_{l+1}} \underbrace{B_{l+1,d-k-1}(t)}_{=0\ \forall\, d,k} p_l^k \tag{5}$$

$$= \sum_{i=l-d+k+1}^{l} B_{i,d-k-1}(t) p_i^{k+1} \tag{6}$$

We now insert $k = 0$ and $k = d$ and get the expected formular by iteration over all $0 < k < d$ as

$$x(t) = \sum_{i=l-d}^{l} B_{i,d}(t) p_i^0 = ... = \sum_{i=l}^{l} B_{i,0}(t) p_i^d = B_{l,0}(t) p_l^d = p_l^d \tag{7}$$