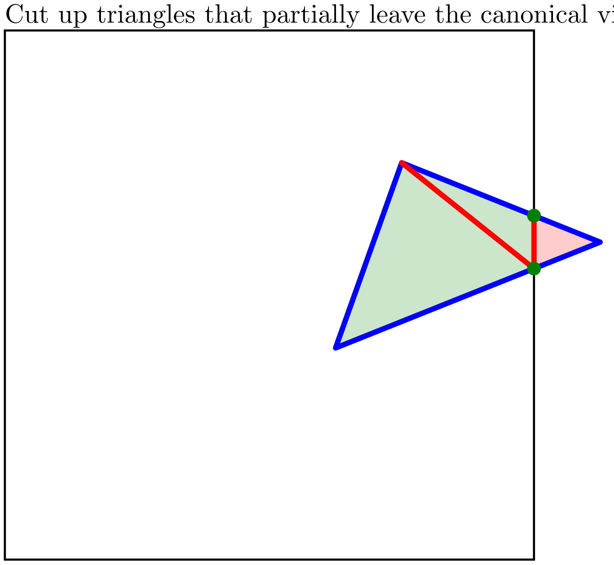
Clipping

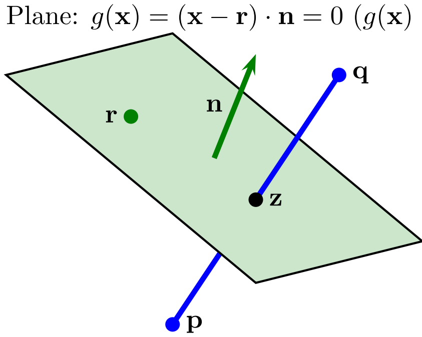
CS 130

1. Goal of clipping
   - Rasterization is rather expensive
   - Involves doing work for every pixel that is inside each triangle
   - Would be nice to avoid doing this work for pixels that cannot be seen
   - Discard triangles that are outside the canonical viewing volume
   - Cut up triangles that partially leave the canonical viewing volume



   - All triangles being rasterized are now fully in the viewing area
   - They might still be behind something.

2. Segment-plane
   - Representation
     - Segment: $f(s) = \mathbf{p} + s(\mathbf{q} - \mathbf{p})$; $0 \leq s \leq 1$
     - Plane: $g(\mathbf{x}) = (\mathbf{x} - \mathbf{r}) \cdot \mathbf{n} = 0$ ($g(\mathbf{x}) > 0$ is outside)



- Cases
  - $g(\mathbf{p}) \leq 0$ and $g(\mathbf{q}) \leq 0$: inside
  - $g(\mathbf{p}) > 0$ and $g(\mathbf{q}) > 0$: outside
  - Otherwise, the segment intersects the plane
- Intersection location
  - Intersection: $\mathbf{z}$
  - On segment: $\mathbf{z} = \mathbf{p} + s(\mathbf{q} - \mathbf{p})$
  - On plane: $(\mathbf{z} - \mathbf{r}) \cdot \mathbf{n} = 0$

$$\begin{aligned}
0 &= (\mathbf{z} - \mathbf{r}) \cdot \mathbf{n} \\
&= (\mathbf{p} + s(\mathbf{q} - \mathbf{p}) - \mathbf{r}) \cdot \mathbf{n} \\
&= (\mathbf{p} - \mathbf{r}) \cdot \mathbf{n} + s(\mathbf{q} - \mathbf{p}) \cdot \mathbf{n} \\
s &= \frac{(\mathbf{r} - \mathbf{p}) \cdot \mathbf{n}}{(\mathbf{q} - \mathbf{p}) \cdot \mathbf{n}}
\end{aligned}$$

3. Triangle-plane
   - Discard triangle if all vertices outside plane
   - Accept triangle if all vertices inside plane
   - Otherwise, need to clip triangle
   - Compute intersection points (segment-plane intersections)
   - Triangulate new region; creates one or two triangles
4. Triangle-box
   - Clip against walls one at a time
   - May produce many triangles