Removing dependencies from `recob::Track`

Gianluca Petrillo

Fermi National Accelerator Laboratory

LArSoft Coordinators' Meeting, March 15th , 2016

The status

- `recob::Track` had two methods:

```
double ProjectedLength(geo::View_t view) const;
double PitchInView
  (geo::View_t view, size_t trajectory_point = 0) const;
```

  that depend on `Geometry` service

- every program using it needs to have the whole LArSoft geometry code and the *art* service infrastructure available

LArSoft recommendations endorse light-weight data products with little code; additional operations can be provided by means external to them (e.g. façade objects).

The change

The two offending methods have been moved into a different library:

repository:	`lardata` (still the same)
header:	`lardata/RecoBaseArt/TrackUtils.h`
source:	`lardata/RecoBaseArt/TrackUtils.cpp`
library:	`lardata_RecoBaseArt`
namespace:	`lar::utils`

Function prototypes:

```
double lar::utils::TrackProjectedLength
  (Track const& track, geo::View_t view);
double lar::utils::TrackPitchInView
  (Track const& track, geo::View_t view, size_t trajectory_point = 0);
```

(note the additional `recob::Track` argument)

The fixes

How to fix:

- `pTrack->PitchInView(view, point)` becomes
  `lar::utils::TrackPitchInView(*pTrack, view, point)`
- `pTrack->ProjectedLength(view)` becomes
  `lar::utils::TrackProjectedLength(*pTrack, view)`
- need `#include "lardata/RecoBaseArt/TrackUtils.h"`
- need `lardata_RecoBaseArt` in `CMakeLists.txt` of the fixed algorithm/module (might be there already)
- a branch has been provided with the fix for `dunetpc` (`feature/gp_NoServiceInTrack`); `uboonecode` does not use those methods

Failure to fix will yield compilation and linking errors.

Reference information in LArSoft breaking change wiki page