It's clear we need to restrict things a little....... 3ci

For any but the smallest sets of clauses the number of resolution steps can be huge
So what can we do to reduce redundancy?

• Recall: at the ground level (no variables) we have a merge operation that removes duplicate literals from a clause.

eg $p \vee \neg q \vee \neg q \vee p \equiv p \vee \neg q$

In other words it simplifies a clause by removing redundant literals.

• The analogous and more general operation is called *Factoring*
• Unlike merge, factoring does not always preserve equivalence

e.g. P(a,x) ∨ P(y,b) factors to give P(a,b), but the two clauses are NOT equivalent

On the other hand factoring is sometimes necessary
eg given ¬P(a) ∨ ¬P(v) and P(x) ∨ P(y)
What resolvents can you form? (Remember to rename variables apart)

• Logically we can derive the empty clause:
¬P(a) ∨ ¬P(v) means ∀v[¬P(a) ∨ ¬P(v)] from which we can derive ¬P(a), and
P(x) ∨ P(y) means ∀x∀y[P(x) ∨ P(y)] from which we can derive ∀z.P(z)
We *factor* by applying a binding to enable literals to be merged.

Herbrand Interpretations

4avi

<u>Some Definitions</u>: Let L be a language for a set of sentences S

The *Herbrand Universe* HU of L is the set of terms using constants and function symbols in Sig(L). **(It is assumed there is always at least one constant.)**

The *Herbrand Base* HB of L is the set of ground atoms using terms from HU

An *Herbrand Interpretation* HI of L is a subset of the atoms in HB;
they are the set of true atoms in **HB = {c: val(c) = true and c is in HB}**

An *Herbrand model* of S is an Herbrand interpretation M of L
that forces val(S) =True for each sentence in S under M

If S has a Herbrand model we say S is *H-satisfiable*. If not S is *H-unsatisfiable*.

NOTE: If Sig(L) includes any function symbols then the Herbrand Universe is infinite.

There is assumed always one constant in HU, so HU $\neq \emptyset$.