

1. the entire stack trace of the error message (not just the last part as above)

```
ABC_irdata_exclude_orphan_dandelion = ir.io.to_dandelion(ABC_irdata_exclude_orphan)
ABC_irdata_exclude_orphan_dandelion
```

```
-----
ValidationError                                Traceback (most recent call last)
~/./.conda/envs/dandelion/lib/python3.8/site-packages/airr/schema.py in validate
e_row(self, row)
    273         try:
--> 274             if spec == 'boolean': self.to_bool(row[f], validate=
True)
    275             if spec == 'integer': self.to_int(row[f], validate=T
rue)
```

```
~/./.conda/envs/dandelion/lib/python3.8/site-packages/airr/schema.py in to_bool
(self, value, validate)
    143         if bool_value is None and validate:
--> 144             raise ValidationError('invalid bool %s' % value)
    145         else:
```

ValidationError: invalid bool T + T

During handling of the above exception, another exception occurred:

```
ValidationError                                Traceback (most recent call last)
/tmp/ipykernel_1432158/721953288.py in <module>
----> 1 vdj_Covid19_HC_HIV_irdata_exclude_orphan_dandelion = ir.io.to_dandeli
on(vdj_Covid19_HC_HIV_irdata_exclude_orphan)
      2 vdj_Covid19_HC_HIV_irdata_exclude_orphan_dandelion
```

```
~/./.conda/envs/dandelion/lib/python3.8/site-packages/scirpy/io/_util.py in che
ck_wrapper(*args, **kwargs)
    65         for i in check_args:
    66             _check_anndata_upgrade_schema(args[i])
--> 67         return f(*args, **kwargs)
    68
    69         return check_wrapper
```

```
~/./.conda/envs/dandelion/lib/python3.8/site-packages/scirpy/io/_io.py in to_da
ndelion(adata)
    738
    739     data = pd.DataFrame.from_dict(contig_dicts, orient="index")
--> 740     return ddl.Dandelion(ddl.load_data(data))
    741
    742
```

```
~/./.conda/envs/dandelion/lib/python3.8/site-packages/dandelion/utilities/_core
.py in __init__(self, data, metadata, germline, distance, edges, layout, grap
h, initialize, **kwargs)
    59
    60         if self.data is not None:
--> 61             self.data = sanitize_data(self.data)
    62             self.n_contigs = self.data.shape[0]
    63             if metadata is None:
```

```

~/conda/envs/dandelion/lib/python3.8/site-packages/dandelion/utilities/_util
ities.py in sanitize_data(data, ignore)
    356
    357     # check if airr-standards is happy
--> 358     validate_airr(data)
    359     return (data)
    360

~/conda/envs/dandelion/lib/python3.8/site-packages/dandelion/utilities/_util
ities.py in validate_airr(data)
    411         contig.update({required: ''})
    412     RearrangementSchema.validate_header(contig.keys())
--> 413     RearrangementSchema.validate_row(contig)
    414
    415

~/conda/envs/dandelion/lib/python3.8/site-packages/airr/schema.py in validat
e_row(self, row)
    276         if spec == 'number': self.to_float(row[f], validate=
True)
    277     except ValidationError as e:
--> 278         raise ValidationError('field %s has %s' %(f, e))
    279
    280     return True

```

ValidationError: field productive has invalid bool T + T

2. the result of ABC_irdata_exclude_orphan.obs.columns

ABC_irdata_exclude_orphan.obs.columns

```

Index(['multi_chain', 'extra_chains', 'IR_VJ_1_c_call', 'IR_VJ_2_c_call',
      'IR_VDJ_1_c_call', 'IR_VDJ_2_c_call', 'IR_VJ_1_consensus_count',
      'IR_VJ_2_consensus_count', 'IR_VDJ_1_consensus_count',
      'IR_VDJ_2_consensus_count', 'IR_VJ_1_d_call', 'IR_VJ_2_d_call',
      'IR_VDJ_1_d_call', 'IR_VDJ_2_d_call', 'IR_VJ_1_duplicate_count',
      'IR_VJ_2_duplicate_count', 'IR_VDJ_1_duplicate_count',
      'IR_VDJ_2_duplicate_count', 'IR_VJ_1_j_call', 'IR_VJ_2_j_call',
      'IR_VDJ_1_j_call', 'IR_VDJ_2_j_call', 'IR_VJ_1_junction',
      'IR_VJ_2_junction', 'IR_VDJ_1_junction', 'IR_VDJ_2_junction',
      'IR_VJ_1_junction_aa', 'IR_VJ_2_junction_aa', 'IR_VDJ_1_junction_aa',
      'IR_VDJ_2_junction_aa', 'IR_VJ_1_locus', 'IR_VJ_2_locus',
      'IR_VDJ_1_locus', 'IR_VDJ_2_locus', 'IR_VJ_1_productive',
      'IR_VJ_2_productive', 'IR_VDJ_1_productive', 'IR_VDJ_2_productive',
      'IR_VJ_1_v_call', 'IR_VJ_2_v_call', 'IR_VDJ_1_v_call',
      'IR_VDJ_2_v_call', 'has_ir', 'clone_id', 'clone_id_by_size',
      'sample_id', 'locus_VDJ', 'locus_VJ', 'productive_VDJ', 'productive_VJ',
      'v_call_genotyped_VDJ', 'v_call_genotyped_VJ', 'd_call_VDJ',
      'j_call_VDJ', 'j_call_VJ', 'c_call_VDJ', 'c_call_VJ',
      'duplicate_count_VDJ', 'duplicate_count_VJ', 'duplicate_count_VDJ_1',
      'duplicate_count_VJ_1', 'duplicate_count_VDJ_2', 'junction_aa_VDJ',
      'junction_aa_VJ', 'status', 'status_summary', 'productive',
      'productive_summary', 'isotype', 'isotype_summary', 'vdj_status',
      'vdj_status_summary', 'constant_status_summary', 'receptor_type',
      'receptor_subtype', 'chain_pairing', 'Disease', 'Sampling_time',
      'Severity', 'clone_id_size', 'cc_aa_alignment', 'cc_aa_alignment_size'],
      dtype='object')

```